Advanced delay reduction algorithm based on GPS with Load Balancing

Vinay Kumar Shukla [1] and Kavita Singh[2]

[1][2]Shri Ramswaroop Memorial College of
Engineering and Management, Uttar Pradesh,INDIA

[1]shuklavinay708@gmail.com
[2]singh.kavita052@gmail.com

Abstract

A Mobile Ad-Hoc Network (MANET) is a self-configuring network of mobile nodes connected by wireless links, to form an arbitrary topology. The nodes are free to move arbitrarily in the topology. Thus, the network's wireless topology may be random and may change quickly. An ad Hoc network is formed by sensor networks consisting of sensing, data processing, and communication components. There is frequent occurrence of congested links in such a network as wireless links inherently have significantly lower capacity than hardwired links and are therefore more prone to congestion. Here we proposed a algorithm which involves the reduction in the delay with the help of Request_set created on the basis of the location information of the destination node. Across the paths found in the Route_reply (RREP) packets the load is equally distributed.

Keywords

MANET, Request_set, RREQ (Route request message), RREP, GPS, Load Balancing Routing protocol.

1. INTRODUCTION

Mobile wireless adhoc networks are built of collection of mobile devics which can communicate through wireless links. Routing is task of forwarding packets from source to destination. Routing is hard in mobile wireless adhoc networks. The algorithm presented here is explicitly designed for use in the wireless environment of an ad hoc network. When a host needs to communicate with destination , it dynamically determines the route based on cached information and on results of route discovery protocol. In this paper, we present the routing algorithm for mobile ad-hoc networks. UP till now many routing algorithms have been proposed to solve the routing problem in mobile adhoc networks. The proposed algorithm performs better for solving routing problems in Mobile Ad Hoc networks. Most of the proposed algorithms use a blind flooding technique during the route discovery process. This method is inefficient and leads to excessive overhead . To overcome this problem, the proposed routing protocol uses a query localization technique that significantly reduces the network traffic and increases the performance of network.The limited resources such as CPU, battery set special challenges in routing design in MANET’s[1].

2. PREVIOUS RELATED WORKS

The Qualities of a good routing algorithm provides loop-free routes, provides multiple routes (to alleviate congestion), and establishes routes quickly (in case of link failure). Different routing protocols have been proposed and are classified into two major categories as Proactive and Reactive [2].

DOI : 10.5121/ijdps.2013.4304

2.1 DSR

DSR is an algorithm used in mobile wireless networks for routing purpose. DSR maintains route caches to accumulate routes that have been established by means of flooding or through promiscuous overhearing [3]. Although it provides single path routing, but , it could be amended to support multipath routing. More significantly, it suffers from a scalability problem due to the nature of source routing. As the network becomes larger, control packets (which collect node addresses for each node visited) and message packets (which contain full source routing information) also become larger [4][5].

2.2 AODV Algorithm

This routing protocol is intended for use by mobile nodes in ad hoc networks when two hosts wish to communicate with each other and a route is created to provide such connection[6].Ad hoc on Demand Distance Vector Routing is a novel algorithm for the operation of ad hoc networks. Each mobile host in the network operates as a specialized router and routes are obtained on demand with little or no reliance on periodic advertisements [7]. The algorithm leads to dynamic , self starting, multihop routing between participating mobile nodes wishing to establish and maintain an ad hoc network[8].

2.3 Location-Aided Routing (LAR)

It utilizes location information to improve performance of routing protocols for ad hoc networks. By using location information, Location-Aided Routing protocols limit the search for a new route to a smaller “request zone” of the ad hoc network [9].

3. PROPOSED PROTOCOL TECHNIQUE

The main objective of our proposed algorithm is to decrease the communication overhead resulted from the redundant exchange of route request(RREQ) messages between all nodes in the network on the basis of the use of location information of the destination node. This goal is achieved by selecting a *subset* of nodes (rather than all nodes) to participate in the RREQ packet forwarding process with help of GPS. For any hop, H_i, the subset of nodes is chosen with a guarantee that the nodes belonging to this set will cover all the nodes in the next hop, H_{i+1} and will be close to the destination. Therefore, along all hops, the nodes will reached via a minimum number of nodes, thus, the message overhead will be reduced as much as possible.

In mobile ad hoc networks, *flooding* [10] is the major technique that is used for communication and message transmission between nodes. Flooding is the process wherein, each node, and upon receiving a particular message, sends the received message again to its neighboring nodes (i.e. the nodes that are positioned within its transmission range). This process continues until all nodes in the network receive the message (at least once). When some source wants to send data to destination, it initiates a path discovery process by sending an *RREQ* message to all of its 1-hop neighbors. In turn, those 1-hop neighbors resend the *RREQ* message to their entire 1-hop neighbors (that is, 2-hop neighbors of node *S*), and so on. This process is illustrated in Figure 1.

The proposed algorithm consists of three steps:-

1. Finding the request set of all nodes
2. According to the GPS, forwarding the Route Request (RREQ) packet to the selected nodes in Request_set.

3. Application of load balancing technique on the selected paths from source to destination

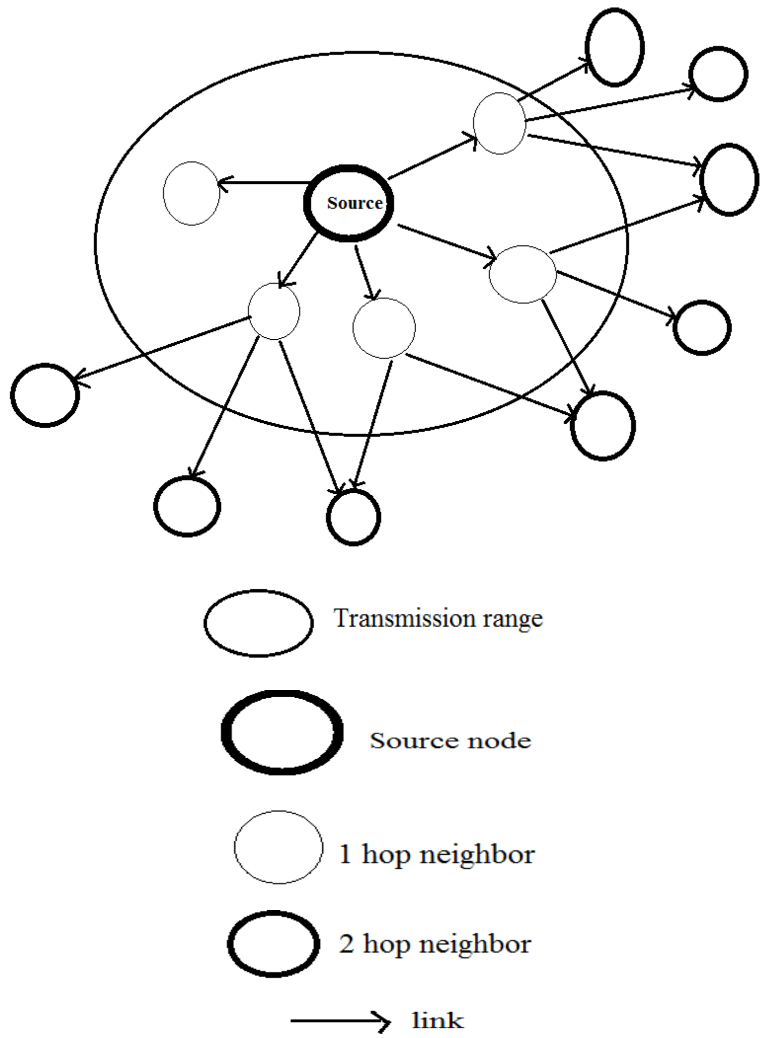

Figure 1 - Working of Flooding based algorithm in forwarding of RREQ packets

3.1 Creation of Request_set of nodes-

Request_Set of node *X* is defined as the set of *X*'s 1-hop neighbors that ensures full coverage for the entire 2-hop neighbors of *X* [11] .It is worthy to mention here that the neighbors of any node are sorted in ascending order based on their *VALUE* (in our algorithm, the value is the velocity or speed of the node). Also it is important to mention that the building of *Request_Set*s is performed in a distributed manner, that is, each node builds its own *Request_Set* independently from any other node. Therefore, at any time *T*, the *Request_Set* of a node *A* is different than the*Request_Set* of a node *B* (unless both nodes have the same neighbors), Moreover, the *Request_Set* of any particular node *N* at time *T1* is different than the *Request_Set* of the same node at time *T2* (this is determined instantly based on the mobility status of nodes, and which node have joined the transmission range of *N* and which have departed).

In any hop, the members of *Request_Set* are chosen carefully such that they provide full coverage of the nodes in the next hop. In addition, they are with lower mobility in comparison with other non-covering set nodes. These two features (i.e. the selection of a **subset** of nodes with **velocity-awareness**) are important strength points of our proposed algorithm, since using *Request_Sets* will reduce message overhead associated with flooding. In addition, allowing nodes with low mobility to participate in path discovery process will enforce stability in the network, since the lower the mobility of nodes, the more stable the links.

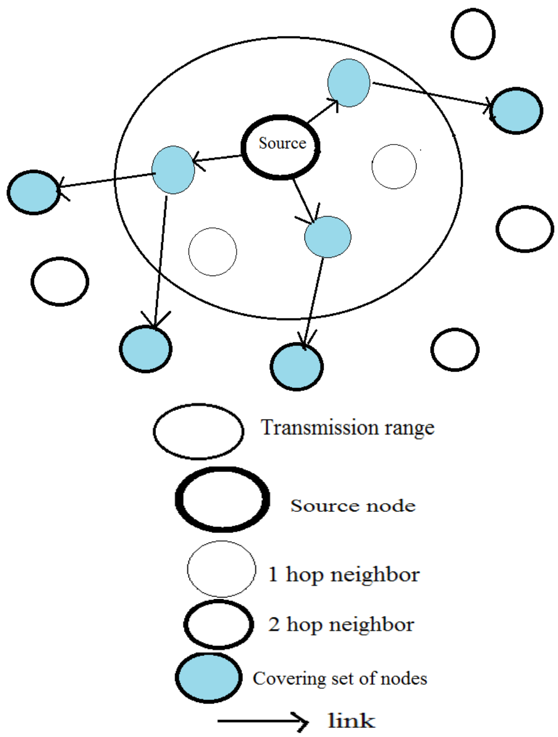

Figure 2 - Working of Request_set based algorithm

When a node *X* wants to send an *RREQ* message it firstly creates its *Request_Set* using its neighbor table; starting from the first entry in the neighbor table (i.e. the neighbor with lower velocity) until reaching full coverage for the 2-hop neighbors. For each neighbor, *X* checks if this neighbor adds additional coverage (i.e., if it has path(s) to some of the 2-hop neighbors that are not covered previously by any of the selected nodes). If so, *X* adds the current neighbor to its *Request_Set* and checks if there are more nodes that are not covered by any node yet; if so, *X* repeats the process for the next neighbor until all 2-hop neighbors are covered.

The algorithm used to build the *Request_Set* is shown

1. START
2. Request_set(x)=null
3. for each node m in nbrtable(x);
4. If m gets additional coverage(i.e it has a path for 2-hop neighbours that are not covered by previously other nodes)
5. Add m to Request_set(x);
6. If all the 2-hop neighbours are covered(i.e reached) by Request_set(x)
7. Return Request_set(x)
8. End

3.1.1 Message Complexity in Flooding-based neighbor Election Algorithms:

If node *S* has *H1* neighbors in its 1-hop transmission range and *H2* nodes in its 2-hop transmission range and *Hi* nodes in its *ith* transmission range, then the number of *RREQ* messages that will be sent by the source node *S* that want to send data to destination will be equal to *H1* messages, which is equivalent to the number of its 1-hop neighbors. Further, each node in the 1-hop transmission range will send *H2* messages for all of its 1-hop neighbors (that is, 2-hop neighbors of node *S*), therefore, the total number of messages that will be sent by nodes in the 1-hop transmission range will be *H1**H2*. This process continues until reaching the last hop *i,* where the number of messages that are sent by *Hi-1* nodes will be *Hi-1 * Hi*, where *Hi* is the number of nodes in the last hop. The message complexity associated with flooding-based forwarding of RREQ packets is illustrated in equation

Message complexity = H1 + H1* H2 + H2 *H3 +….+(Hi -1) *Hi

3.1.2 Message Complexity in Request Set-Based Routing Algorithm:

If node *S* has *H1* neighbors in its 1-hop transmission range and *H2* nodes in its 2-hop transmission range and *Hi* nodes in its *ith* transmission range, then the number of nodes in the Request_set for these hops will be *H1-C*, *H2-C*, *H3-C*, … and *Hi-C*, respectively. Where *C* is the number of nodes that are excluded from the Request_sets and that will not participate in RREQ packets forwarding. The number of *RREQ* messages that will be sent by the source node *S* that will be only *H1-C,* where the number of messages sent by the Request_set of the first hop will be *H2-C*, which is equivalent to the number of nodes of the Request_set in the second hop.

Message complexity= H1- c +H2 –c +H3 - c + …… +Hi –c

3.2 Reduction of Request_set of each and every node on the application of GPS based algorithm

The technique proposed is related to the Location Aided Routing (LAR) algorithm [9]**.**The location information of a source and destination node is piggy-backed with each route request and route reply packet respectively. During the route localization, firstly, the source node inspects its location cache for any previous route to the destination. The probability of route to destination available in source node's cache is high if it has either communicated with the destination previously or acted as router for it. If the route entry is found, the positional information of the destination (its *x* and *y* coordinates) is used to calculate the distance to it using Equation

$$D_{sd} = +\lambda$$

Dsd represents the distance from the source to the destination node.

Δx and Δy is the difference between the *x* and *y* coordinates of the source and destination nodes respectively. λ is a factor that takes into account the approximation of the distance measure and is given by Equation

$$\lambda = vX(t_c - t_d)$$

t_c is the current time and t_d is the timestamp of the location information. v is the specified maximum speed that a node can move.

i. The distance calculated by the source node D_{sd} is included in the route request packet broadcast to its Request_set. The timestamp of the location information (t_{sd}) used to calculate D_{sd} is also one of the fields in the packet.
ii. When a intermediate node between the source and destination receives the packet for forwarding , the first action it takes is to query its cache for an entry of route to the destination. If an entry is found, the timestamp of entry (t_{id}) is compared with the location information timestamp in RREQ packets(t_{sd}) . If t_{id} is newer than t_{sd}, it implies that the intermediate node's location information for the destination is more recent or fresh. there does not exist any stale route to the destination.
iii. The intermediate nodes in Request_set then determines the distance from source to destination through itself .

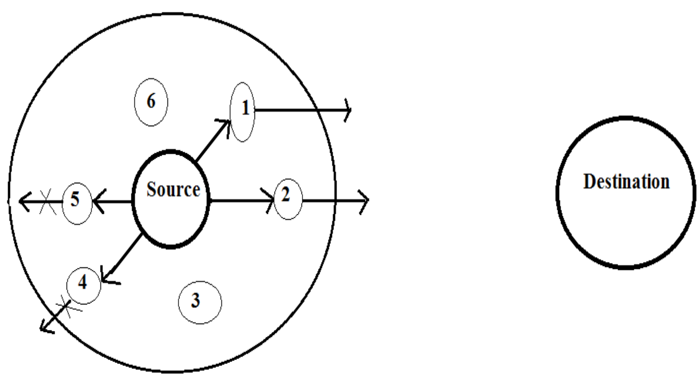

Figure 3- Working of GPS algorithm

iv. The distance field in the route request packet is updated according to the intermediate node.
v. The intermediate node then calculate its distance to the destination node and compares this value with the source to destination distance.
vi. If D_{sd} is found to be larger than D_{id} (the intermediate node is closer to the destination than the node from which the route request arrived), the intermediate node further rebroadcasts the route request packet to its request set.
vii. If this condition is not met the packet is dropped, as shown in figure 3. Although the nodes 1, 2, 4,5 are in the Request_set of source but only nodes 1, 2 rebroadcast the RREQ packets to its neighbor on the basis of locational information of destination. Nodes 4, 5 are far away from destination as compared to nodes 1, 2.
viii. During the localization of route request process ,if any intermediate node has no entry for the destination in its location cache , it does not execute the route localization algorithm. The node instead broadcasts the route request to all nodes in its Request_set as it is done in partial flood in Figure 2.
ix. If a route cannot be identified using the partial flooding it is assumed that destination is unreachable and route discovery is terminated. [13].

3.3 Load Balancing across the various detected routes in Route_Reply packets

3.3.1 Load Balancing Technique

CWF(Combined Weight Function) = (a * Energyi + b * PR) / (c*Leni * d*TLi)
Where a,b,c and d are constants used to normalize the metrics,
i).The priority of route i related to residual energy level (Energyi),

$$Energy_i = Energy_{res}(i)/(N_i * Energy_{initial})$$

where ,$Energy_{initial}$ is the node's energy in the beginning of simulation (which is set to the same value for all nodes),

$Energy_{res}$ (i) is the total of the remaining energy in the nodes of route i.

N_i is the number of nodes in the route i

ii).(Traffic load of route i)

$$TL_i = \frac{\sum_{n=1}^{N_i} R1_n + \sum_{k=1}^{NH_n} R2_n^k}{NH_i \times C_{\max}}$$

where, $R1_n$ = number of routes via node n

NH_n = number of neighbors for node n,

$R2_n^K$=number of routes through kth neighbor of node n,

NH_i = number of neighbors of nodes in route i which has Nn nodes (repetitive neighbor is taken into account once),

C_{max} = maximum connection which a node can establish in a network (which is set to the same value for all nodes)

iii).Length which indicates the priority of route i regarding the length of the route is defined as, Leni = $ALen_i$/MLen

where, $ALen_i$ is the actual length of route i (i.e. number of hops in route i) and MLen is the maximum length that a route can take in DSR routing protocols.

iv). the signal strength received for free-space propagation is measured by receiving node [14]

$$Pr=Pt(\lambda/4\pi d)^2 G_T G_R$$

λ is wavelength of the carrier, d is distance between sender and receiver.

G_T and G_R are unity gain of transmitting and receiving omni-directional antennas, respectively [15].

Consider min w to be the probability of minimum combined weight function (CWF) of a path which is indexed as k where k =1,2……p_{max} *and* p_{max} is the total number of available paths**,** w_{max} **is** the probability of maximum combined weight function (CWF) of the path and $\mathbf{w_{op}}$ is the probability of weight function of other paths.

- w_{min} implies no information reaches to the destination.
- w_{max}= 1- w_{min} implies all the information reaches correctly to the destination.
- The probability w_{max} or w_{min} of a path is independent of the probability w_{op} since there are no common nodes for these paths.

- A packet containing T bits of information is to be transmitted in such a way that the probability w_{max} of the path p_{max} must be maximized.
- Extra U bits are added to the original information of T bits for enhancing reliability of message transmission with the help of source coding technique.
- Hence, S = T+U is treated as one network layer packet.
- Original T bits are split into L blocks of equal size b and U bits are split into M blocks of same size b as of T bits.

The original message can be reconstructed if at least L blocks out of M blocks reach the destination using L for M diversity coding. This is achieved by using Lagrange interpolation and secret sharing scheme [16]. The blocks of S bits information can be judicially distributed over the available paths.

An **allocation vector** is defined as $Av = [A_{vj}]$, where, A_{vj} is the number of equal size blocks distributed over the path j.

If p is the number of paths available, then the allocation vector has the form as

$A_v = (A_{v1}, A_{v2} \ldots\ldots\ldots A_{vp})$ where $p <= p_{max}$

Since the block size is b, we can write,

$S = b\sum_{j=1}^{n} A_{vj}$ and S = r*T So, r =S/T

Where r = overhead factor.

If b_j is the number of blocks that reaches the destination through the path j, then

$p(b_j = A_v) = w_{max}$
$p(b_j = 0) = w_{min}$

Assuming for w_{max} , all the allocated blocks over the path will reach to the destination successfully and in case w_{min} , all the blocks sent over the path is lost.

The w_{max} in terms of p and A_v is given by

$$w_{max}(p, A_v) = p\left\{\sum b_j \geq \frac{\sum_{j-1}^{n} A_{vj}}{r}\right\}$$

Where $\sum_{j-1}^{n} A_{vj}$ total number of blocks by which the information S bits is fragmented.

The probabilities of combined weight function whether minimum or maximum is not same for all paths.

The paths with maximum probability cannot be assigned with fewer blocks than a path with a minimum probability,
Therefore, $w_{min} <= w_{min} + 1$ *follows* $A_v >= A_v + 1$

We assume the uniform A_v without the loss of generality i.e. all the paths will be assigned with equal number of blocks though they have different probability [17].

4 Experimental Results

On the basis of comparison between the AODV protocol and our proposed protocol ,we discover following –

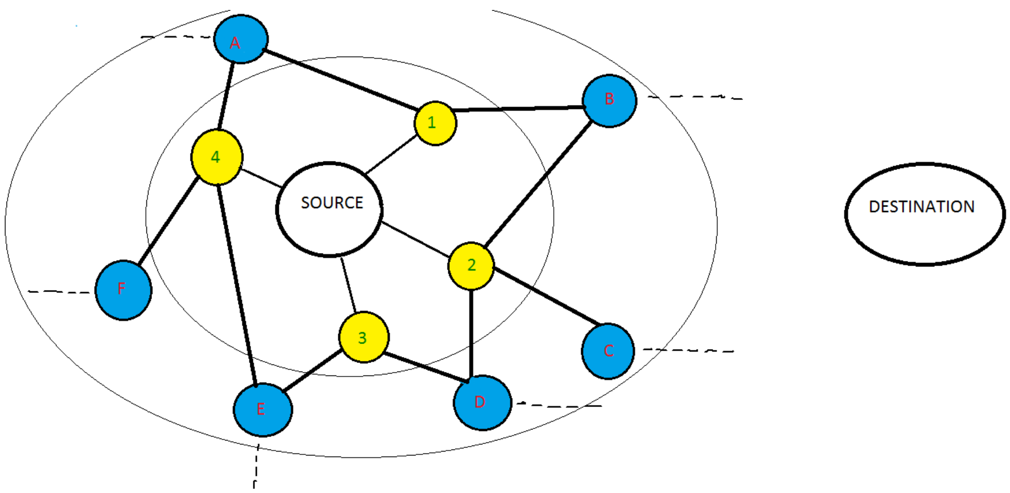

Figure 4- Example of a MANET

In fig. 4 the source needs to send the packets to the destination. firstly, it discovers the route to the destination with help of forwarding of RREQ packets to the neighboring nodes.

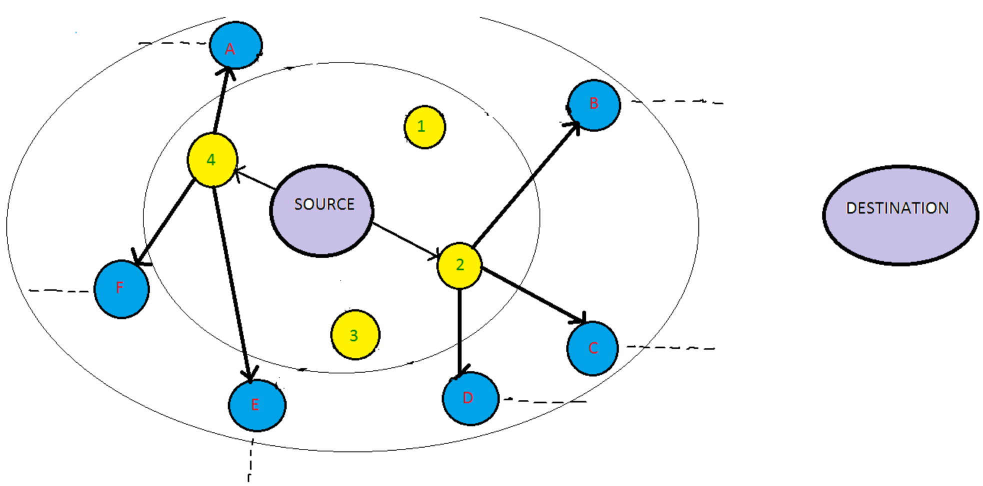

Figure 5- flooding in MANET of RREQ packets during route_discovery

Traditionally many of the algorithms in MANET perform flooding of RREQ packets during the route_discovery process to the destination.The fig. 5 consists of source with nodes 1, 2 ,3, 4 as its 1-hop neighbors. While nodes A, B, C, D, E, F are the 2-hop neighbors of the source.

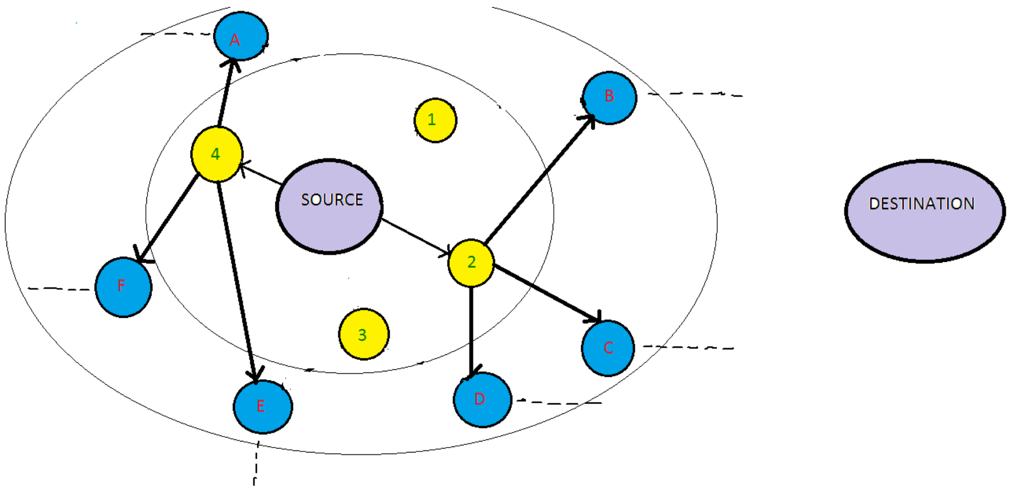

Figure 6- Application of Request_set algorithm to the neighbors of the source

In fig. 6 , the transmission ranges of various nodes are-
Node 1 – A, B
Node 2-B , C ,D
Node 3- E, D
Node 4- A, E, F

After the application of Request _set algorithm , The Request_set of source contains nodes 2 and 4 only , because they are able to cover all the 2- hop neighbors of the source.

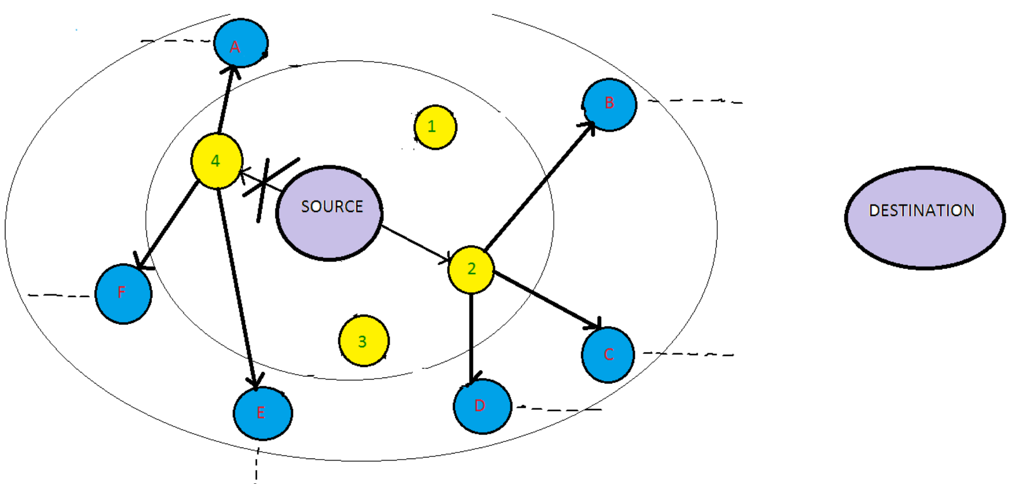

Figure 7- Application of GPRS based algorithm on the Request_set of the source node

In fig. 7, the Request _set of source node contains node 2 and node 4. on the basis of GPRS based algorithm, the location of source, destination and intermediate node are taken into account.As node 2 is closer to the destination as compared to node 4, so only node 4 receives the RREQ packet to further forward it to its neighbors until destination is reached.

Thus, following conclusion arises:-

i. AODV is based on total or full flooding of RREQ packets to all the neighbors of the source while our protocol involving the use of Request_set and GPS leads to partial flooding of RREQ packets to only few neighbors of the source.
ii. AODV takes more time in transmission of packets and increases the overhead also while our protocol does not.
iii. AODV does not focus on load balancing while our protocol does. The proposed protocol equally distributes the load to all the routes in RREP packets
iv. Hence our protocol is efficient than the AODV protocol.

5 Conclusion

In this paper, we propose an Effective Route discovery and Load balancing Technique for Mobile Ad hoc Networks. In this technique, initially, a request_set for each and every node is calculated and further the Request_sets of nodes are refined by the Global Positioning System (GPS) algorithm. In the GPS based algorithm, the location information of source and destination nodes are piggybacked in the Route_request packets. In load balancing, a combined weight function based on route length, traffic load, energy level and freshness for each route is estimated and stored in the route cache. We select best n paths based on their combined weight value among the selected paths in Route_reply packets. Then traffic has to be distributed over these paths. The algorithm results in reduction in the delay, overhead and energy while increasing the packet delivery ratio, when compared to AODV.

References

[1] Sunsook Jung, Nisar Hundewale and Alex Zelikovsky, "Node Caching Enhancement of Reactive Ad Hoc Routing Protocols", Proceedings IEEE Wireless Communication and Networking Conference (WCNC'05), March 2005, Volume 4, 1970- 1975.
[2] D. Lang, A Comprehensive Overview about Selected Ad Hoc Networking Routing Protocols, Master Thisis, Technische Uni. Munich, Germany, 2003.
[3] Alvin Valera, Winston K.G. Seah and SV Rao, "Cooperative Packet Caching and Shortest Multipath Routing in Mobile Ad hoc Networks", In Proceedings of IEEE INFOCOM, , pp. 260-269, March-April, 2003.
[4] David B. Johnson, David A. Maltz and Josh Broch, "DSR: The Dynamic Source Routing Protocol for Multi-Hop Wireless Ad Hoc Networks", In Ad Hoc Networking, edited by Charles E. Perkins, Chapter 5 (2001), pp. 139-172, 2001.
[5] K.Santhi, Dr. M.Punithavalli "A Reliable and Effective Cache Management Technique for DSR Protocol in Mobile Ad hoc Networks", In the Proceedings of 3rd International conference on Computer Modeling and Simulation (ICCMS 2011), Vol.2, pp 198-204.
[6] C. Perkins, E. Belding and S. Das, Ad hoc on-demand distance vector (AODV) routing, Request for Comments: 3561, 2003.
[7] Charles E. Perkins and Elizabeth M. Royer, "Adhoc On Demand Distance Vector Routing", Proceedings of the 2nd IEEE Workshop on Mobile Computing Systems and Applications, pp. 90-100, New Orleans, LA, February 1999.
[8] D. Lang, A Comprehensive Overview about Selected Ad Hoc Networking Routing Protocols, Master Thisis, Technische Uni. Munich, Germany, 2003.
[9] Young-Bae Ko and Nitin H. Vaidya, "Location-Aided Routing (LAR) in mobile ad hoc networks", Wireless Networks, Volume 6 , Issue 4, Pages: 307 – 321, July, 2000.
[10] Abdalla MH, Aamir S, Irfan A, Mike W. "Dynamic Probabilistic Flooding Performance Evaluation of On-Demand Routing Protocols in MANETs CISIS '08 Proceedings of the 2008 International Conference on Complex, Intelligent and Software Intensive Systems, IEEE,pp. 200-204, 2008
[11] Ala'a N. Alslaity "Stable Neighborhood-Based Route Discovery Protocol For Mobile Ad Hoc Networks", Dissertation, Jordan University of Science and Technology. May, 2012.

[12] Shobha.K.R and K. Rajanikanth, "Intelligent Caching in on-demand Routing Protocol for "Mobile Adhoc Networks", World Academy of Science, Engineering and Technology 56, 2009.

[13] IMPROVING PERFORMANCE OF MOBILE AD HOC NETWORKS USING EFFICIENT TACTICAL ON DEMAND DISTANCE VECTOR (TAODV) ROUTING ALGORITHM International Journal of InnovativeComputing, Information and Control ,Volume 8, Number 6, June 2012

[14] S. Harous, M. Aldubai, Q. Nasir," An Energy Aware Multi-path Routing Algorithm",International Conference on Communication, Computer & Power (ICCCP'07)

[15] R. Vadivel, "An Adaptive Reliable Routing Protocol for Mobile Adhoc Networks", 2010 International Conference on Informatics, Cybernetics, and Computer Applications (ICICCA2010).

[16] Adi Shamir,"How to share the secret", Volume 22, Issue 11 (November 1979).

[17] Reliable and Effective Load Balanced Routing Protocol for Mobile Ad Hoc Networks European Journal of Scientific Research ISSN 1450-216X Vol.81 No.3 (2012), pp.357-366

AUTHORS

Vinay Shukla

I am undergraduate pursuing B.Tech from Shri Ramswaroop Memorial College of Engineering and Management from Department of Computer Science and Engineering. I have keen interest in the research area of NETWORKING. So, I along with my project partner is presenting a new algorithm for reducing the delay in networking. I am also thankfull to my project guide "Mr.Ashish Kumar".

Kavita Singh

I am undergraduate pursuing B.Tech from Shri Ramswaroop Memorial College of Engineering and Management from Department of Computer Science and Engineering.. So, I along with my project partner is presenting an advanced algorithm for reducing the delay in transmission of packets from source to destination based on GPS and Load Balancing . I am also thankfull to my project guide "Mr.Ashish Kumar" for providing his guidance in our project.

.