A Novel Congestion Avoidance Based Load Balanced Routing with Optimal Flooding in Mobile Ad hoc Network

P. P. Tandon and Bibhudatta Sahoo

National Institute of Technology Rourkela, Orissa
E-mail : premppt@gmail.com, bdsahu@nitrkl.ac.in

ABSTRACT

The QoS of Mobile Ad hoc Network is mostly affected by the congestion at any intermediate node in a selected routing path. In this paper, we propose a congestion avoidance basedload balanced routing scheme for mobile ad hoc network. Each node keeps track of the number of data packets transmitted byhim as well as the data packets transmitted by its one hop neighboring nodes along with their flag bit status for the currenttime interval. The proposed approach is attempts to avoid thecongestion of a node by selecting the disjoint paths. This is achieved by setting a flag bit with the time limit TTL, at the node. On exceeding of this value, the flag bit is reset. By this approach we have attempted to limit the flooding and congestionof the node along with effective balancing of the traffic load.

Keywords: *congestion, interference, Flooding, Load balanced routing, MANET.*

1. INTRODUCTION

Mobile ad hoc network is a resource constrained randomly deployed network, in which almost all the nodes are battery constrained. Data communication in such a network is possible along multiple hops, nodes that are in communication range of each other can communicate directly, while the nodes that are spatially disjoint uses other intermediate nodes to relay the packets. Application of such a network is in personal area networking, emergency operations such as search and rescue in disaster caused areas, policing and fire fighting, military applications such as on the battle field, civilian environment such as taxi cab networks, meeting rooms, sports stadium etc. Routing in such a mobile infrastructure less network where the topology of the network keeps on varying frequently is a challenging task due to lack of central control. Much of the earlier work have been focused on routing in such an infrastructure less environment like AODV, DSR, DSDV, ZRP etc.[14]. These protocols are categorized as table driven routing protocols such as DSDV, WRP etc. and on demand routing protocols such as AODV, DSR, TORA, etc. Some hybrid protocols such as ZRP have also been proposed which combines the best features of above mentioned protocols. Performance comparisons of above mentioned protocols shows that the on demand routing protocols outperforms the table driven routing protocols. While the other set of work have been propagated in energy saving mechanism. But almost all of the above mentioned routing protocols in one way or other tries converge into shortest path routing. One of the advantages of using shortest path routing is that it is good for overall energy efficiency because energy needed to transmit a packet is directly proportional to path length or number of hops [11]. But the shortest path routing is restricted to use the same set of hops to route the data packets. This approach leads to exceeding the load of some nodes in the routing path and thus causing some of the nodes to die earlier resulting into holes in the network or even worst into partitioning of the network. Thus the need for load balanced routing emerges. Load balancing problem in mobile ad-hoc network is NP-hard problem [11]. Load balanced routing algorithm can be divided into two categories:

1. Based on cost function used for selecting the path.
2. Based on the modification over routing algorithm.

1. Based on the cost function used: in this category load balanced routing algorithm can be classified as Traffic size based load balanced routing and delay based load balanced routing. In Traffic size based load balanced routing load of a node is defined as the number of packets transmitted by the node and tries to balance this per node traffic load. Examples of such a load balancing strategies are Dynamic load aware routing (DLAR) [3], Load balanced ad hoc routing (LBAR) [2], load sensitive routing (LSR) [4] etc. While the delay based routing protocols takes delay as the path computation metric and tries to avoid selecting the congested nodes during path formation. Examples of such load balanced routing strategies are Delay oriented shortest path routing protocols (DOSPR) [5], Load aware on demand routing protocols (LAOR) [6] etc.

2. Based on the modification: in this category load balanced routing scheme can be classified as AODV based or DSR based. The load balanced routing scheme that are based on AODV routing protocol as a bench mark, are: *(i) Dynamic Load Aware Routing Protocol (DLAR) [3], (ii) Load Balance Ad Hoc Routing Protocol (LBAR) [2]*, and *(iii)Load Aware On Demand Routing Protocol (LAOR) [6]*. While the second group of load balanced routing protocols that are build over DSR as a bench mark are (i)Load Sensitive Routing (LSR) [4], (ii) Free Degree Adaptive Routing (FDAR) [20], and (iii) Accumulative relative Delay Load Balanced Routing Protocol (ARDLB) [1].

The rest of the paper is organized as follows. Section 2 presents the related work on different routing protocols on Mobile Ad hoc Network. A mathematical model for wireless net work is described in section 3. The load balance routing problem is presented as an optimization problem. The section 4 describes the proposed congestion avoidance based load balanced routing scheme. The performance of proposed scheme over shortest path routing is verified through simulation are presented in Section 5. Finally, conclusions and directions for future research are discussed in Section 6.

2. RELATED WORK

Among the various routing algorithm only the associativity based routing (ABR)[19] considers the load as a routing metric. The ABR considers load as the second routing metric hence the protocol does not account for various traffic loads for each session. In dynamic load aware routing (DLAR) [3] load information is defined as the number of packets that exists in the interface of a node. In DLAR nodes need to keep informing the destination about the load information by piggybacking them on data packets so that the destination always have the latest load information for deciding the path. In routing with load balancing scheme(LBAR) [2], the destination collects as much information as possible to choose the optimal route in terms of minimum nodal activity(i.e the number of active routes passing by the node). By gathering the nodes activity degrees for a given route the total route activity degree is found. In load sensitive routing (LSR) [4] the load information depends on two parameters: *total path load* and *the standard deviation of the total path load*. In LSR if an intermediate node has multiple route request stored in its cache for the same session that were collected from dropped duplicate packets, it will replace the rest of the path in the route reply packets with the best path in terms of load destination monitor’s the load cost for each incoming data packets during an active session and informs the source when the load costs exceeds certain percentage. If so found new route is better than the earlier route then the source immediately adopts the new route. In load aware on demand routing (LAOR) [6] the total path delay is considered as the load metric for calculating the route. LAOR measures each node’s delay, which includes the contention and transmission delays. LAOR uses current node delay, previous node delay and the number of packets being queued in the current node to determine the congested node in the path. These congested nodes drop any incoming route requests; intermediate nodes after receiving route requests updates the load information in the packets and resends them to all neighbors. The intermediate node will also check the load value and will drop the packet, if its load value is worst than the currently stored one, or if better than, the packet will be resent. At destination any new coming route request carrying a new path will be compared with the currently active path. If better in terms of load then this path will be sent to the source as route reply. In accumulative relative delay load balanced routing protocol (ARDLB) [1] a new technique has been used for load calculation which can utilize relative delay in the node queue instead of the node delay alone. In ARDLB [1] each node calculates its own delay at each packet arrival and stores it in appropriate memory space moreover it updates part of the load information in the passing by packets. The path carried by any packet has the total delay as stamp. In network aware MAC and routing protocol for effective load balancing in ad hoc wireless networks with directional antenna [15] makes efficient use of directional antenna and define a new parameter called route coupling which takes care of the data loss occurred by path coupling (interference) and resolves it by the finding out the zone disjoint routes. In a novel delay oriented shortest path routing protocol for mobile ad hoc networks (DOSPR) [5], the access contention delay at the MAC layer is computed and factorized to the total delay computation. This delay computed is used as the *path selection metric*. In load balanced routing considering power conservation in wireless ad hoc networks[7] energy threshold value is computed and transmitted with route request. This energy threshold value is utilized as the path selection criteria over the network. In load balanced routing through virtual paths [8] load value is defined to be number of packets in bytes and thus defines the parameters regional load and total load. Based on these parameters balances load of the defined region and total path load. In load balanced short path routing in wireless ad hoc networks [11] four greedy methods have been defined for effectively balancing the load of the particular path by taking the number of packets delivered by any node as its load parameter. Recently in interference aware load balanced routing in wireless mesh network (IALBR) [10] data loss by the interference is taken into account. In IALBR probability of channel busy is defined as the load parameter for the path selection criteria. In (FDAR) [20] free degree of the nodes is defined as the load parameter, which is calculated as the ration of transmission rate of the node to the receiving rate of the node.

In this paper, we propose a new load balanced routing scheme that can efficiently manage network load in an efficient way. This protocol is based on traffic size and an adaptation from the class of DSR based load balanced routing protocol. In this protocol we define the number of bytes of packets node has transferred and the number of active neighboring host that are active for that particular time instant as the load parameter. This scheme balances the traffic load as well as reduces the data coupling. However in our scheme intermediate nodes can not send route reply back to the source even if it has route to the destination. As the cache information may not give the accurate load information of the listed route. Load of the neighboring nodes is known by the periodic broadcast of hello messages.

3. NOTATIONS AND DEFINITIONS

Wireless nodes can be modeled as set of points S in a plane let n denote the number of points in the plane we assume that the communication range of each node is one. The communication graph of S is then, a unit disk graph $U(S) = G(S, E)$ where $(p, q) \in E$ if the Euclidean distance between $p, q \in S$ is at most 1. Let ρ denotes the set of paths satisfying set of route requests R. Then for any $r \in R$ the load $l(v)$ incurred to any node v is defined to be sum of the number of packets (in bytes) delivered by that node

$$L = \sum_{r \in R} l_r$$

The path load is defined as the sum of all the load of the respective intermediate nodes contributing the path i.e. for any $P \in \rho$

$$PL = \sum_{i \in P} L_i$$

Let K be the transmission range of a node $v \in S$ then $K + \Delta k$ be its interference range. We say that two paths are disjoint when they are at least $K + 2 * \Delta k$ Euclidean distance apart. Let T denotes the time period, we define the activity of the node as for any time $t \in T$, node $v \in S$ is actively participating in the routing activity i.e. either transmitting or receiving, for reflecting such an activity of the node we use a flag bit.

$$Aflat = \begin{cases} & \textit{if node is active at time } t \\ & \textit{otherwise} \end{cases}$$

Activity flag (*Aflag*) is set, with a value of time to live (*TTL*), if for the current fraction of time t, the node $v \in S$ is actively participating in routing; otherwise it is reset. The value *TTL* is assigned as the twice of the time taken by the route request message to reach the destination node. Each time the node receives data packets the time to live field is updated with this *TTL* value. We define the problem of load balanced routing as

$$\min imize \sum_{i \in P} L_i$$

Subject to *Aflag*

4. CONGESTION AVOIDANCE BASED LOAD BALANCED ROUTING (CALBR)

Routing in ad hoc networks, according to on demand fashion, occurs in two steps in first step meta path is formed based on flooding and in second the destination node selects the best path among all the available paths. We assume that each node keeps track of the information of its one hop neighbor such as amount of load relayed by the node i.e. number of packets delivered by it, and its flag bit status with its *TTL* value. Now the algorithm is defined as follows:

4.1 Information Exchange

Each node periodically broadcasts hello message to its one hop neighboring nodes consisting its node ID, the number of packets transmitted by it and its flag bit information with its *TTL* value. This hello message is limited to rebroadcast again by the receiving nodes. The node receiving this hello message will record the values contained in the hello message, failing to receive hello message from earlier listed node denotes the node is no longer connected and link is no longer valid and hence the its recorded value is deleted from the routing table. Similarly receiving hello message from new node denotes new link has been formed and its received values must be recorded in routing table.

4.2 Algorithm

4.2.1 At source node

Initially source node does not have the location information of destination node so it broadcasts route request (RREQ) message for the route discovery. The route request message carries the source ID, destination ID, and a path vector which contains the relaying node ID, and amount of the traffic the relaying node has delivered. This RREQ is again forwarded by the neighboring nodes till the destination node has been reached; this mechanism is known as flooding. After broadcasting route request message Source node waits for route reply packets till predefined amount of time. Once this amount of time has expired after broadcasting route request message and no route reply packet has received the source node again broadcasts the route request message. Once the source node receives route reply packet it comes to know that a route has been build and starts transmitting data packets via received route.

4.2.2 At intermediate node

When the intermediate node receives this RREQ packets it first checks its routing table weather its any of the neighboring node is active for that instant of time if yes then it drops the route request packet, else adds its node ID, load information in the path vector of the route request packet and again re broadcast it.

When intermediate node receives route reply packet it first updates its routing table by making flag bit 1 (set) with the TTL value listed in the route reply packet along with the cache information and then unicast this route reply packet and waits for the data packets to arrive. When the intermediate node receives data packets it first replaces the TTL value by the $ttl amount which is defined as the same value of TTL listed in the route request packet received earlier. Once this $ttl time is out the flag bit is reset.

When intermediate node relays a data packet, then for every data packet being relayed it updates its routing table by the respective load information and $ttl value being defined.

4.2.3 At the destination node

Once the destination node receives the route request message it first records the TTL value defined in the route request packet for that respective route and waits for predefined amount of time to collect other route request messages after collecting the various route's information, the destination node then chooses the best node whose path load is least among all the available paths. The destination node then simply swaps the path vector of the chosen route and after attaching the calculated $ttl value in the time field of route reply packet sends the route reply message and simultaneously set its flag bit.

When intermediate node receives this route reply packets, it first set its activity flag i.e. *Aflag* = 1 then after recording the information in its route cache, unicast this route reply packet to the next hop node defined in the path vector of the route reply packet. When the source node receive this unicast route reply (RREP) packet it comes to know that a route has been build and then first set its flag bit and then starts data transmission.

When any of the intermediate relaying node moves away from the transmission range of its upstream neighbor due to mobility, its upstream neighbor informs the source node by sending the route error (RERR) message. The source node upon receiving RERR message again floods the RREQ message to obtain the optimal route and process of route construction is repeated. For example consider the network model below :

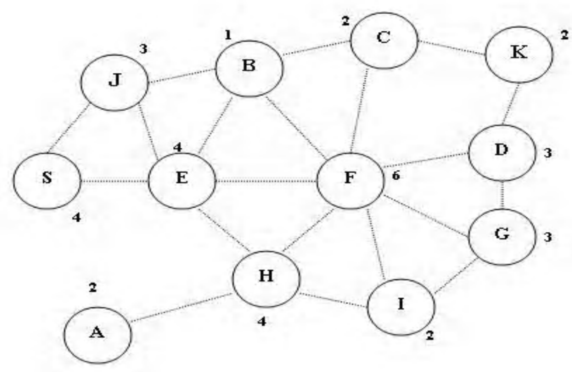


Fig. 1: Network model

In fig. 1 the virtual links represents the nodes that are in transmission range of the respective nodes or simply the transmission range of the respective nodes and the weights among the nodes represents the amount of traffic the respective nodes has delivered. Suppose the source node A wants to send data packets to destination node D then node A broadcasts route request message this route request message is heard by node H as it is the only node in the transmission range of node A. so node H after attaching its load value and flag status of the neighboring nodes available at current instant rebroadcasts it to its neighboring node i.e. node E, node F and node I, where the procedure is repeated. Lastly when the destination node D receives this route request message it calculates the path load which is the sum of the load values of all the intermediate nodes that has attached its ID in the route request packet. the destination node then sets its flag bit and reverses the path vector contained in route request message and sends it as route reply. The intermediate node when receives this route reply first sets its flag bit and then unicast it. When the source node receives this route reply it comes to know that a route has been build and then starts data transmission. For example fig. 2 shows the route constructed between source destination pair node A and node D.

The following figure below shows the status of routing tables at the respective nodes when source node A has sent two bytes of data packets to destination node D. note that we have use # sign in the type field to reflect the node ID belongs to the

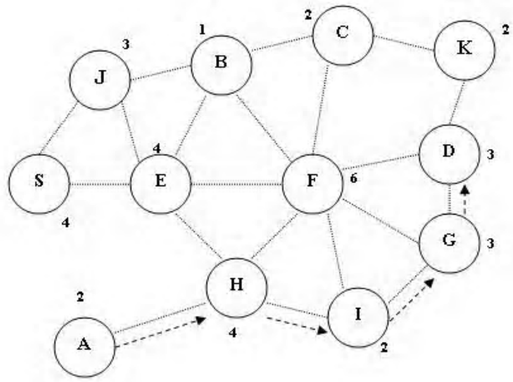


Fig. 2: Illustration of route construction

Routing Table at node A

SID	DID	CACHE
A	D	D-G-I-H-A

NID	Type	LOAD	AFlag	TTL
A	#	4	1	$ttl
H	$	6	1	$ttl

Routing Table at node H

SID	DID	CACHE
A	D	D-G-I-H-A

NID	Type	LOAD	AFlag	TTL
H	#	6	1	$ttl
A	$	4	1	$ttl
E	$	4	0	-
F	$	6	0	-
I	$	4	1	$ttl

Routing Table at node I

NID	DID	CACHE
A	D	D-G-I-H-A

NID	Type	LOAD	AFlag	TTL
I	#	4	1	$ttl
H	$	6	1	$ttl
E	$	4	0	-
F	$	6	0	-
G	$	5	1	$ttl

Routing Table at node E

SID	DID	CACHE

NID	Type	LOAD	AFlag	TTL
E	#	4	0	-
S	$	4	0	-
J	$	3	0	-
B	$	1	0	-
F	$	6	0	-
H	$	6	1	$ttl

Routing Table at node F

SID	DID	CACHE

NID	Type	LOAD	AFlag	TTL
F	#	6	0	-
E	$	4	0	-
B	$	1	0	-
C	$	2	0	-
D	$	3	1	$ttl
G	$	3	1	$ttl
I	$	4	1	$ttl
H	$	6	1	$ttl

Routing Table at node G

SID	DID	CACHE
A	D	D-G-I-H-A

NID	Type	LOAD	AFlag	TTL
G	#	5	1	$ttl
I	$	4	1	$ttl
F	$	6	0	-
D	$	3	1	$ttl

Routing Table at node D

SID	DID	CACHE
A	D	D-G-I-H-A

NID	Type	LOAD	AFlag	TTL
D	#	3	1	$ttl
G	$	5	1	$ttl
F	$	6	0	-
K	$	2	0	-

Routing Table at node S

SID	DID	CACHE

NID	Type	LOAD	AFlag	TTL
S	#	4	0	-
J	$	3	0	-
E	$	4	0	-

Routing Table at node J

SID	DID	CACHE

NID	Type	LOAD	AFlag	TTL
J	#	3	0	-
S	$	4	0	-
E	$	4	0	-
B	$	1	0	-

Routing Table at node B

SID	DID	CACHE

NID	Type	LOAD	AFlag	TTL
B	#	1	0	-
E	$	4	0	-
J	$	3	0	-
C	$	2	0	-
F	$	6	0	-

Routing Table at node C

SID	DID	CACHE

NID	Type	LOAD	AFlag	TTL
C	#	2	0	-
B	$	1	0	-
F	$	6	0	-
K	$	2	0	-

Routing Table at node K

NID	DID	CACHE

NID	Type	LOAD	AFlag	TTL
K	#	2	0	-
C	$	2	0	-
D	$	3	1	$ttl

Fig. 3: Status of routing tables at respective nodes when source node A has sent 2 bytes of data packets to destination node D.

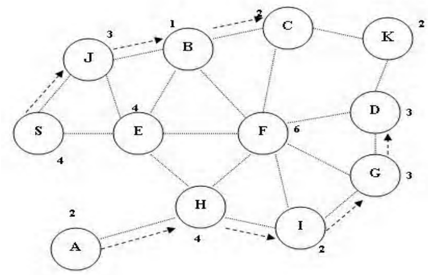


Fig. 4: Illustration of route construction between node pair S and D

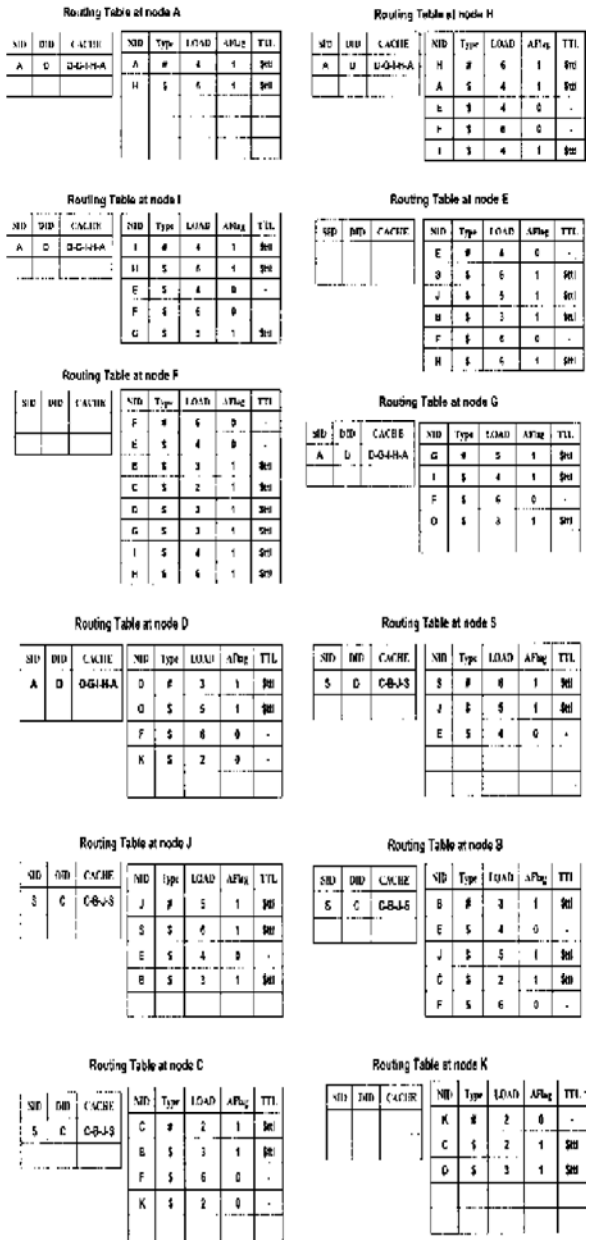
respective node and $ sign to reflect the node is a neighbor node. $ttl represents the calculated *TTL* value for flag bit. At the same instant while communication is going on between node A and node D suppose node S wants to communicate with node C. then according to the algorithm and the status of routing tables at the respective nodes, node C selects the route C-B-F-S. Note that the route request message will not be forwarded by node E as at that interval of time the flag bit of its neighboring node H will be set.

The figure below illustrates the status of the updated routing tables at the respective nodes at the condition when source node A has send two bytes of data packets to destination node D and the source node S has sent data packet of size two bytes to the destination node C.

5. PERFORMANCE ANALYSIS

In our experiment set of nodes were randomly deployed in the 500×500 grid and each node has initial velocity of 10 m/s moving accordingly random way point model and each node has transmission range of 200 m. After rigorous analysis and simulation we present our comparison result with the shortest path routing strategy for one set of source destination pair operating at aligned traffic of ten packets per second.

Above comparison shows that our algorithm can effectively balance the traffic load, in fact the maximum load on any network node denotes the lifetime of the ad hoc network as each node is battery constrained and the amount of packets to be transmitted by any node is directly proportional to its battery power consumed. Following figure shows the variation of lifetime of this network.



Routing Table at node A

SID	DID	CACHE
A	D	D-G-I-H-A

NID	Type	LOAD	AFlag	TTL
A	#	4	1	$ttl
H	$	6	1	$ttl

Routing Table at node H

SID	DID	CACHE
A	D	D-G-I-H-A

NID	Type	LOAD	AFlag	TTL
H	#	6	1	$ttl
A	$	4	1	$ttl
E	$	4	0	-
F	$	6	0	-
I	$	4	1	$ttl

Routing Table at node I

SID	DID	CACHE
A	D	D-G-I-H-A

NID	Type	LOAD	AFlag	TTL
I	#	4	1	$ttl
H	$	6	1	$ttl
E	$	4	0	-
F	$	6	0	
G	$	5	1	$ttl

Routing Table at node E

SID	DID	CACHE

NID	Type	LOAD	AFlag	TTL
E	#	4	0	-
S	$	6	1	$ttl
J	$	5	1	$ttl
B	$	3	1	$ttl
F	$	6	0	-
H	$	6	1	$ttl

Routing Table at node F

SID	DID	CACHE

NID	Type	LOAD	AFlag	TTL
F	#	6	0	-
E	$	4	0	-
B	$	3	1	$ttl
C	$	2	1	$ttl
D	$	3	1	$ttl
G	$	3	1	$ttl
I	$	4	1	$ttl
H	$	6	1	$ttl

Routing Table at node G

SID	DID	CACHE
A	D	D-G-I-H-A

NID	Type	LOAD	AFlag	TTL
G	#	5	1	$ttl
I	$	4	1	$ttl
F	$	6	0	-
D	$	3	1	$ttl

Routing Table at node D

SID	DID	CACHE
A	D	D-G-I-H-A

NID	Type	LOAD	AFlag	TTL
D	#	3	1	$ttl
G	$	5	1	$ttl
F	$	6	0	-
K	$	2	0	-

Routing Table at node S

SID	DID	CACHE
S	D	C-B-J-S

NID	Type	LOAD	AFlag	TTL
S	#	6	1	$ttl
J	$	5	1	$ttl
E	$	4	0	-

Routing Table at node J

SID	DID	CACHE
S	C	C-B-J-S

NID	Type	LOAD	AFlag	TTL
J	#	5	1	$ttl
S	$	6	1	$ttl
E	$	4	0	-
B	$	3	1	$ttl

Routing Table at node B

SID	DID	CACHE
S	C	C-B-J-S

NID	Type	LOAD	AFlag	TTL
B	#	3	1	$ttl
E	$	4	0	-
J	$	5	1	$ttl
C	$	2	1	$ttl
F	$	6	0	-

Routing Table at node C

SID	DID	CACHE
S	C	C-B-J-S

NID	Type	LOAD	AFlag	TTL
C	#	2	1	$ttl
B	$	3	1	$ttl
F	$	6	0	-
K	$	2	0	-

Routing Table at node K

SID	DID	CACHE

NID	Type	LOAD	AFlag	TTL
K	#	2	0	-
C	$	2	1	$ttl
D	$	3	1	$ttl

Fig. 5: Status of the updated routing table

6. CONCLUSION

In this paper we demonstrated the congestion avoidance based load balanced routing scheme that can efficiently reduce the data collision caused by route coupling. By this method we can reduce the packet loss due to interference; also we demonstrated

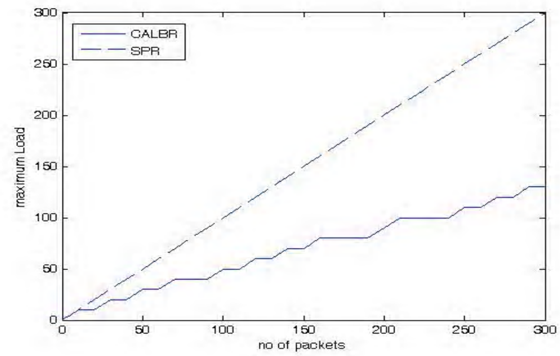


Fig. 6: performance comparison of proposed scheme with the routing scheme that uses shortest path routing (SPR) as route selection metric

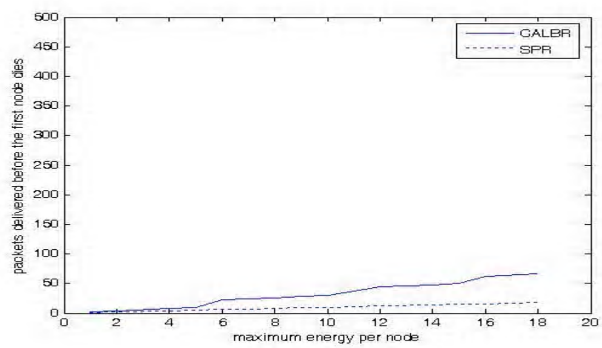


Fig. 7: performance comparison of proposed scheme with shortest path routing (SPR) in terms of lifetime of network

this method may be used to limit the excessive flooding. Our work is still in progress, we are working on a new scheme that will use traffic load and hop count together as a route selection metric for better performance. A further investigation may be made to handle the MAC layer contention with provision for route adaptation and maintenance.

REFERENCES

1. Omar al-jarrah & Musab Salem Al-Hadrusi "Load Balanced Ad hoc Routing Protocol" in proceedings of the first mobile computing and wireless communication international conference, 2006, MCWC 2006, september 2006
2. Hossam Hassanein, Audrey Zhou, "Routing with Load Balancing in Wireless Ad Hoc Networks," Proceedings of the 4th ACM international workshop

on Modeling, analysis and simulation of wireless and mobile systems; 2001:89- 96.

3. Sung-Ju Lee, Mario Gerla, “Dynamic Load-Aware Routing in Ad Hoc Networks,” Proceedings of The Third IEEE Symposium on Application-Specific Systems and Software Engineering Technology (ASSET); 2000.
4. Kui Wu, Janelle Harms, “Load-Sensitive Routing for Mobile Ad Hoc Networks,” Proceedings of the Tenth International Conference for Computer Communications and Networks; 2001 :540-546.
5. S. T. Sheu & J Chen “A Novel Delay Oriented Shortest Path Routing (DOSPR) Protocol For Mobile ad-hoc Networks” Proceedings of IEEE ICC’01, Helsinki, Finland, June 2001
6. Joo. Han Song, Vincent Wong, Victor C. M. Leung, “Load-Aware On- Demand Routing (LAOR) Protocol for Mobile Ad Hoc Networks,” The 57th IEEE Semiannual 2003; 3:1753- 1757.
7. Huen Kyuong Cho, Eun Seok Kim, & Dae Wook Kang “A Load Balancing Routing considering Power Conservation in Wireless ad hoc Networks” Proceedings of the Sixteenth International Workshop on Database & Experts System Applications IEEE 2005.
8. Abdulrahman H. Altalhi & Golden G Richard “Load Balanced routing Through Virtual Paths: Highly Adaptive & Efficient Routing Scheme for Ad-hoc Wireless Networks” Proceedings of the First International Workshop On Mobile Peer to Peer Computing (MP2P’04), Orlando, FL, USA March 14-17, 2004
9. Yashar Ganjali, Abtin Keshavarzian, “Load Balancing in Ad Hoc Networks: Single-Path Routing vs. Multi-path Routing,” Stanford University; 2003. (Available from: www.ieeeinfocom.org/2004/Papers/23_4).
10. Jing Feng, Ran Xia & Huaibei Zhou “Interference Aware Load Balanced Routing in Wireless Mesh Networks (IALBR)” Proceedings of IEEE May 2007
11. Jie Gao & Li Zhang “Load Balanced Short Path Routing in Wireless Ad-hoc Networks.” Proceedings of IEEE INFOCOM’04 2004.
12. 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 *Proceedings of ACM MOBICOM’98*, Dallas, Texas, USA, October 1998.
13. P. Johansson, T. Larsson, N. Hedman, B. Mielczarek, And M. Degermark, “Scenario-based Performance Analysis of Routing Protocols for Mobile Ad-hoc Networks,” in *Proceedings of ACM/ IEEE MOBICOM’99*, Seattle, Washington, USA, August 1999.
14. T. Larsson, N. Hedman, “Routing Protocols in Wireless Ad-Hoc Networks – A Simulation Study,” Stockholm: Luleå University of Technology. Master Thesis. 1998.
15. Siuli Roy & et al “A Network-Aware MAC and Routing Protocol for Effective Load Balancing in Ad Hoc Wireless Networks with Directional Antenna” in proceedings of the 4 ACM international symposium on mobile ad hoc networking & computing; 2003: pp. 88- 97
16. Byung-jae kwak & Nah-oak song “A Mobility Measure for Mobile ad hoc Networks” in proceedings of IEEE communications letters, vol. 7, no. 8, august 2003.
17. Xavier perez-costa, Christian bettstetter and Hannes hartenstein “Toward a Mobility Metric for Reproducible and Comparable Results in ad hoc Networks Research” Mobicom 2003
18. Fucai yu, Younghwan choi, Soochang park, Ye tian & Sang-ha kim “Stability Oriented Overlay Multicast for Mobile ad hoc Networks” in proceedings of international conference on wireless communications, networking and mobile computing, 2007, WICOM 2007.
19. Chai-keong toh “Associativity based Routing for ad hoc mobile networks” in proceedings of wireless personal communications, SPRINGERLINK 103-139, november 2004.
20. XiaoRan Wang, Shigaeki Tagashira, and Satoshi Fujita “FDAR: A Load-Balanced Routing Scheme for Mobile Ad-hoc Networks” Springer- Verlag Berlin Heidelberg 2007. pp. 186-197.