

---

# A Direction-Based Make-Before-Break Routing Protocol for Vehicular Ad Hoc Networks

USMAN HASSAN\*, FAHIM GOHAR AWAN\*, AND SUHAIL AFTAB QURESHI\*

**RECEIVED ON 27.11.2015 ACCEPTED ON 16.02.2016**

## ABSTRACT

In this paper, we present a novel “Make-Before-Break” routing protocol for VANETs (Vehicular Ad Hoc Networks) which brings stability in route even if it contains some nodes moving opposite to the rest. Few of the existing routing protocols contain the mechanism to prevent route disruption caused by link breakages between oppositely moving nodes in a partitioned network. In our protocol, a node does not forward a RREQ (Route Request) received from a node moving in opposite direction unless a special request is made by that node in RREQ packet. The node which accepts this special request and forwards the RREQ, despite moving opposite to the previous node is called complier node. The complier node adds its information in the RREQ packet. On receiving RREQ, the destination knows that complier node is moving towards some successive nodes in the routing path. Through the RREP (Route Reply), it informs those nodes about the complier node. Each of these successive nodes waits for the approaching complier node. When it comes near, each establishes connection with the complier node on its turn. Thus the route from source to destination is maintained. The proposed protocol achieves packet delivery ratio of 85% under very large speed variation condition among vehicles’ speed.

**Key Words:** Vehicular Ad Hoc Networks, Partitioned Network, Node, Switch Count.

## 1. INTRODUCTION

**V**ANETs have become an active area of research nowadays [1]. Besides numerous safety applications, many entertainment applications of VANETs are being proposed [2-3]. Dedicated data communication between two vehicles on the road requires a fixed multi-hop route between them. Extremely high mobility of vehicles on highways makes the task of providing stable route very challenging. Sometimes when node density is not high, a VANET may get partitioned. In this case, all the vehicles transferring data are not

moving in same direction. However, links between oppositely moving nodes break more quickly. Many existing reactive routing protocols attempt to achieve route stability by employing selectivity during route setup. Few of them contain the mechanism to prevent route disruption caused by link breakages between oppositely moving nodes in the route.

Almost all of routing protocols for VANETs utilize the position and velocity of vehicles obtained from increasingly available facility of GPS (Global Positioning

---

\* Department of Electrical Engineering, University of Engineering & Technology Lahore.

System) [4-7]. Some of these position-based routing protocols use a fixed route to transfer data from source to destination while the others just “push” the data towards the destination [8-9].

A GPSR (Greedy Perimeter Stateless Routing) protocol has been proposed in [10] where each node forwards the data packet to the node closest to the destination among its neighbors. Sometimes, at bends on the roads, a node may find no node closer to the destination than itself. Perimeter routing is employed in such situations i.e. the prescribed node forwards the data to the first of its neighbors in a particular direction, say clockwise. This protocol is not very promising in VANETs where routing holes needing perimeter routing or some other recovery mechanisms frequently occur lowering the performance [11-12].

A broadcasting protocol SIFT (Simple Forwarding over Trajectory) is presented in [13], in which data packets are broadcast after appending trajectory information to them. If a node not located within the indicated geographical path happens to receive the packet, it drops it. All other receiving nodes initiate countdown timers. The initial value of timer depends on distance from the previous node. The far the distance from previous node, the lower is the initial value of timer. The node whose timer reaches zero first, broadcasts the data packet. The other nodes stop their timers and discard the data packet after hearing this broadcast. A major defect in the above protocol is the delay caused by setting timers [14].

Authors in [15] introduced a scheme to select an optimal route based on the expected lifetimes of individual links. Observing that on the highways, typically a vehicle stays on the same lane for an exponentially distributed amount of time, it derived equations to compute the time any two vehicles with different speeds are likely to stay in the radio range of each other and found that optimality criterion for a stable route allows only

monotone change in speeds of successive intermediate nodes. This is not a complete protocol and assumes that all vehicles including the destination are moving in same direction [16-17].

A Movement Prediction-based Routing Protocol for Vehicle-to-Vehicle Communications has been proposed in [18] where selective nodes with small differences in their speeds are chosen for the route and route remains intact for at least a predetermined amount of time unless some disruption at PHY level causes a link in the route to break early. One of the methods in our algorithm uses this procedure for node selection. This protocol is silent on the methodology required for route maintenance in case where a node finds only oppositely moving nodes in its radio range and selection of none of those can satisfy the criterion or route maintenance for the specified time.

Some protocols deal with VANET routing in areas where traffic is sparse. But the traffic density considered therein is permanently very low and nodes often don't find any neighbor moving in either direction [19]. They have to store data packets till they come across some nodes.

None of the previously published protocols known to authors, has the procedure to enhance lifetime of routes when network is partitioned in one direction only under normal traffic conditions. [20] observed that in VANETs on highways, there are expected 10 partitions in one direction per 10 km (the expectation of partitions in both directions creating a sparse network is of course less) if on the average, a vehicle has radio range of about 200m [21]. The objective of this work is to develop a reactive routing protocol for VANETs in a highway environment which can:

- (i) Ensure long-time route maintenance between source and destination involving some nodes moving opposite to them by replacing a soon-to-break portion of route in small time with a new sub-route before breakage.

- (ii) Detect loss of data packets because of weakened PHY conditions in time and replace the route like other protocols using fixed routes do.

Our proposed protocol uses packet formats similar to those in AODV (Ad Hoc On-Demand Distance Vector) routing, which gives reasonable performance at low and moderate mobility rates in mobile ad hoc networks [22]. The designed protocol was tested in Qualnet 3.9 simulating VANETs on a highway with varying speeds of vehicles, the maximum being 200 km/h. Results show that our protocol outperforms AODV in terms of packet delivery ratio and delay, managing to save time spent in route discoveries.

## **2. MAKE BEFORE BREAK ROUTING**

An efficient VANET routing protocol exploits the predictability of mobility pattern of vehicles which is often possible with considerable accuracy, thanks to the constraints of road geometry. This section presents a new routing protocol for VANETs on highways, named “A Direction-Based Make-Before-Break Routing Protocol for Vehicular Ad Hoc Networks.” In this protocol, propagation of a RREQ is done only by nodes having same direction of motion as its originator, provided the availability. A special node called Complier node may be moving oppositely to the rest of nodes in the route. Each of those nodes, during continuous data transmission, manages to establish connection in quick time with that oppositely moving Complier node which was not its next hop in the route earlier on. Two methods for RREQ dissemination are presented for the proposed protocol. One is the broadcast method in which source and intermediate nodes broadcast the RREQ and multiple RREQs reach the destination node which responds to the one arriving earliest and ignores the rest. Second is node selection method proposed by [18]. Each node unicasts the RREQ to a node which is likely to remain in its radio range for a certain period of time. In both methods, each node periodically broadcasts Hello messages containing its location and velocity information.

## **2.1 Principles of Operation**

The rules or principles on which our protocol operates are as follows.

- (i) Our protocol named as Make-Before-Break protocol is a reactive routing protocol i.e. a source node broadcasts RREQs for a destination node only when it needs to send data to that destination node.
- (ii) All nodes broadcast periodic Hello messages (Hello message is a short message containing control information) to announce their current locations, velocities, and directions. The interval is 2 seconds. Consequently, every node also hears Hello messages from its neighbors and maintains a table that contains most recent location, speed and direction of each of its neighbor nodes (the nodes within its radio range).
- (iii) Not all the nodes rebroadcast a RREQ upon hearing it. Using some parameters in a RREQ message, the source node or any subsequent node in the routing path can select or indicate the nodes that can rebroadcast that RREQ.
- (iv) Preference is to allow only those nodes to rebroadcast a RREQ which are moving in the same direction as the previous node. In the special case when a node (say X) receiving a RREQ finds no other node moving in the same direction in its radio range, it makes a special request to the oppositely moving nodes to further propagate the RREQ. To do so, X sets the parameter “Please” in the received RREQ to 1 before broadcasting it.
- (v) Another parameter Switch Count within the RREQ keeps count of the changes of directions of the moving vehicles involved in RREQ propagation. Maximum two changes are allowed.

- (vi) Destination node issues RREP (Route Reply) message on reception of RREQ. RREQ traversal path and RREP traversal path are the same. This is possible because during RREQ propagation, each node receiving the RREQ stores the route to source node and forwards RREP along that route when it receives RREP. Note that RREP is unicast node to node (and not broadcast) until it reaches the source node.
- (vii) The destination node becomes aware of all changes of directions of moving vehicles in the routing path on inspecting the RREQ. Using this information which is crucial for the implementation of Make-Before-Break scheme explained later, it is able to indicate through RREP which sequences of nodes in the routing path are moving towards each other.

## 2.2 Broadcast Method

**RREQ Format:** Table1 shows the format used in RREQ messages in the proposed protocol. This structure is an extension to RREQ message structure used in AODV routing protocol. D is a Boolean variable which represents the direction of motion of the node generating or forwarding a RREQ. It is TRUE when the vehicle is moving in the direction of increasing longitude and FALSE otherwise. Prop Dir stands for direction of propagation

of RREQ. If a node sets Prop Dir to 1 in the RREQ it generates or forwards, it means this RREQ is meant for the vehicles having current longitude greater than this node has, while the vehicles having current longitude smaller than this node's should ignore this RREQ. Opposite is the case when a node sets Prop Dir to 0. In this paper, we shall take direction to the right of page as direction of increasing longitude and direction to the left of page as direction of decreasing longitude. If Prop Dir is set to 2 by any node, it means there is no restriction on propagation of RREQ with respect to location of vehicles and any node can forward this RREQ. P stands for Please bit. It is a Boolean variable which is set to TRUE by a node when it wants an oppositely moving vehicle to forward the RREQ transmitted by it. Switch Count is a counter (maximum value = 2) that indicates how many times P bit has been set to TRUE since the RREQ was originated by the source node. The node which sets the P bit to 1 is called Switcher and the oppositely moving node which accepts this request to forward RREQ is called Complier.

**Generating and Forwarding Route Requests:** When P bit is never set to 1. Consider the scenario depicted in Fig. 1(a). In order to communicate with D, S has to first issue a RREQ. It does not know if D is in front of it or behind it. Therefore, it sets Prop Dir = 2 (unspecified) in the RREQ. Switch Count is set to 0 and P is set as FALSE

TABLE 1. RREQ FORMAT IN PROPOSED PROTOCOL

Type	P	J	R	D	G	U	Prop Dir	Hop Count
RREQ ID							Switch Count	
Destination IP Address							Destination Sequence Number	
Source IP Address							Source Sequence Number	
Source Velocity							Source Location	
Forwarder's Velocity							Forwarder's Location	
Switcher 1's Velocity & Location & Hop Count to Source							Complier 1's Velocity & Location	
Switcher 2's Velocity & Location & Hop Count to Source							Complier 2's Velocity & Location	

because S is supposed to be receiving periodic Hello messages from its neighbors. 7 ignores this RREQ as it is moving towards left while S is moving towards right (and P equals FALSE). 1 and 17 have to rebroadcast this RREQ. S had not specified the Prop Dir but 1 can do so. It compares its location with the location of S contained in received RREQ. 1 is on the right of S, therefore, it sets Prop Dir = 1 in RREQ packet and rebroadcasts it. 17 performs exactly the same procedure while 13 sets Prop Dir = 0 before broadcasting RREQ as it is on the left of S. The RREQ packet broadcast by 1 is heard by 2, 17, 8, and 9. As Prop Dir is 1 and 17 is on the left of 1, 17 ignores the RREQ. 8 and 9 too ignore the RREQ because of having travelling direction opposite to that of 1. But 2 rebroadcasts the RREQ.

In this way, some RREQs reach D which sends a RREP in response to the first of them. Exact format of the route reply is given in next section but here it will suffice that D sends RREP to S with empty Be Alert For field.

**When P Bit is Set to 1 by One of the Nodes in RREQ Propagation Path:** Now consider the scenario shown in Fig. 1(b). Node S broadcasts RREQ for Node D with Switch Count = 0 and Prop Dir = 2. It sets Switch Count = 0 because it knows that in its radio range, there is another node (1) travelling in the same direction (thanks to periodic Hello messages broadcast by 1). Node 1 receives this RREQ. Now, node 1 has not received any Hello message from another node (apart from S) moving in the same direction for a while. Therefore, RREQ must be propagated forward by one node or a sequence of nodes moving in direction opposite to node 1 now. So, Node 1 sets P bit = 1, Switch Count = 1, (and Prop Dir = 1 as before) in the received RREQ. Node 1 also puts its location, velocity and hop count to node S in the RREQ (Node 1 is Switcher 1 as per Table 1) and rebroadcasts it. Node 2 receives this RREQ with P bit = 1 and comes to know that there is scarcity of vehicles moving to the right. Complying with the request made by node 1, node 2 further broadcasts

the RREQ despite moving oppositely to node 1. Switch Count is maintained as 1 and P bit is set to 0 again because node 2 knows that another node, Node 3, is moving in the same direction as itself and our protocol does not allow unnecessary switching of moving directions of vehicles in the RREQ propagation path. Node 2 also puts its location and velocity in the RREQ (Node 2 is Complier 1 as per Table 1) before broadcasting. Node 3 receives the RREQ broadcast by node 2. It knows that there is no node moving in the same direction as itself so it rebroadcasts RREQ after setting P bit = 1, incrementing Switch Count to 2 and putting its location, velocity and hop count to S (Node 3 is Switcher 2 as per Table 1). In this manner, RREQ reaches node D.

**RREP Format:** Table 2 shows RREP format used in proposed protocol. This too is an extension to the RREP structure in AODV. We here introduce the new fields briefly. The fields of Be Alert For, From Hop, and To Hop are filled by destination while Switch Count can be changed (albeit increased only) by any node which relays RREP. Be Alert For is the address of a node (a Complier node) which is moving oppositely to a segment of nodes the RREQ forwarded by the latest of which it had entertained. From Hop and To Hop together define a number range. The nodes with their Hop Count to D falling within this range have to remain ready to make connection with the node indicated by the address in Be Alert For field on reception of a Hello message from it.

Approaching Node To Dest Hops field contains the number of hops between this Complier node and the destination. To understand the concept of Switch Count, refer to Fig. 2. Initial value of Switch Count is 0. A node, issuing or forwarding a RREP, increments the Switch Count by 1 before forwarding the RREP if, and only if, the next hop node is moving oppositely to it. Note that a node knows the moving direction of all its neighbors by virtue of periodic Hello messages being broadcast by all nodes.

A node receiving the RREQ also makes a routing table entry for itself which tells it how to reach source node S if it requires so in future. This is why some nodes are involved in RREQ propagation in Fig. 1(b) and RREP forwarding in Fig. 2. Now, on reception of RREQ from node 1, node 2 also stores the route to source node S as per Table 3, besides rebroadcasting the RREQ. Node 2 stores in its routing table entry that in case it has to send a message back to node S, the next hop is node 1. Thus, when node 2 receives RREP from node 3, it consults its routing table for route to node S and comes to know that the next hop is node 1 and Hello message most recently received by node 1 informs it that node 1 is moving

opposite to node 2. Therefore it increments the Switch Count. Same happens for nodes 4 and 3 in Fig. 2.

**The Procedure for Route Maintenance:** Consider the scenario in Fig. 3. We have supposed direction of

TABLE 2. RREP FORMAT IN PROPOSED PROTOCOL

Type	R	A	Switch Count	Hop Count
Destination IP Address			Destination Sequence Number	
Destination Location			Destination Velocity	
Source IP Address			Lifetime	
Approaching Node To Dest Hops			Be Alert For (Node Address)	
Alert Beginning Hop			Alert Ending Hop	

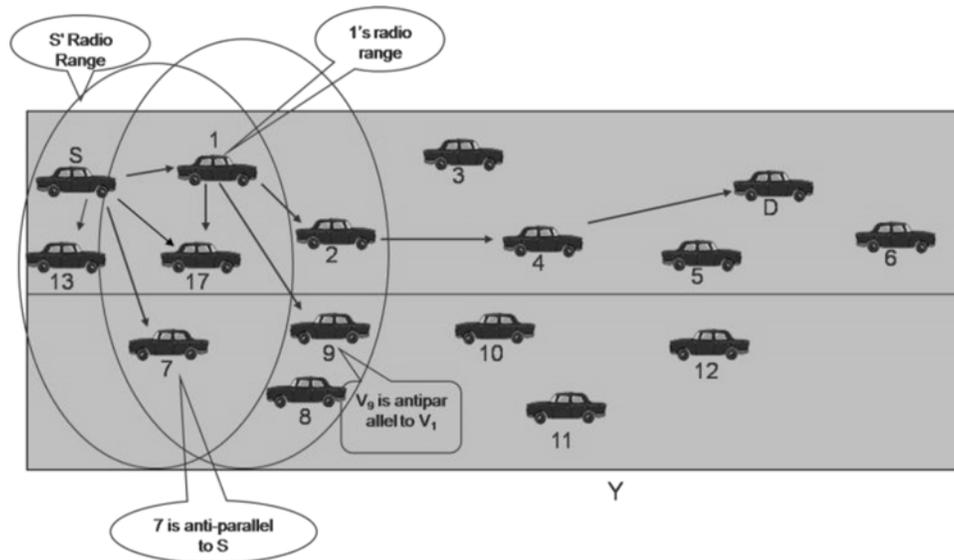


FIG. 1(a). RREQ PROPAGATION IN PROPOSED PROTOCOL WHEN P BIT IS NEVER SET TO 1

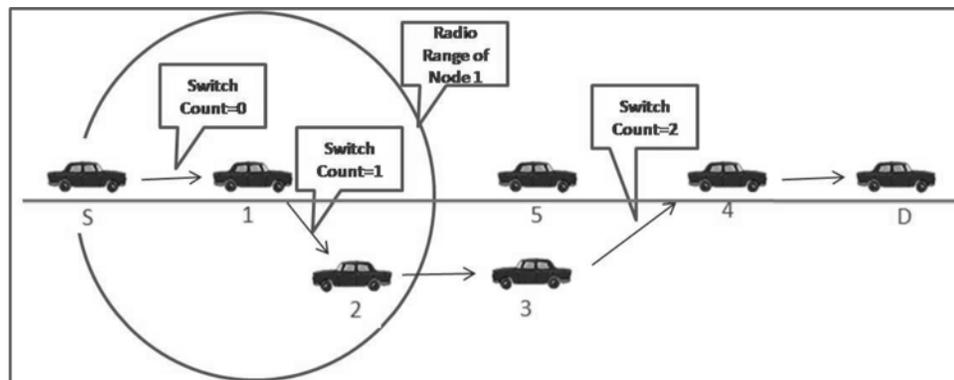


FIG. 1(b). RREQ PROPAGATION IN PROPOSED PROTOCOL WHEN ONE OR MORE NODES SET P BIT TO 1

destination opposite to the rest of nodes for the sake of easy comprehension. Node density here is high enough so that S and all the intermediate nodes forwarding RREQ have neighbors on their right side, moving in the direction same as theirs. Therefore, none of 1, 2, and 3 sets P to TRUE. Consequently, nodes 5, 6, and 7 ignore the RREQs if they hear them. As in AODV, every intermediate node stores route to S in its cache. We added the field of Switch Count in routing table entry. In Table 3, we can see the routing table entry for destination S made by 3 in its routing table. Now D, being the destination, issues RREP. As it is moving oppositely to 3, it increases the received Switch Count value by 1 and caches routing table entries for S and 3, both with Switch Count = 1 as shown in Table 3. In the RREP, D puts Switch Count = 1 in accordance with the procedure defined in the previous section. Table 3 shows the route inserted by S when RREP reaches it. D is moving nearer to S, 1, and 2. It will cross each one of

these in future. Consider a possible situation of future in Fig. 4. If our protocol just confines the RREQ traversal to 1, 2, and 3 having same direction, the fraction of route up to 3 will of course remain stable for a long time but this stability will not be significant since last link in the route (3→D) will break very soon. These breakages during continuous data transmission cause data loss as again and again, route is lost after rediscovery.

TABLE 3. ROUTINGTABLES OF NODES ON RECEIVING RREQ/RREP

Route Storing node	Message Received	Destination	Next Hop	Switch Count
3	RREQ	S	2	0
D	RREQ	S	3	1
D	RREQ	3	3	1
S	RREP	D	1	1

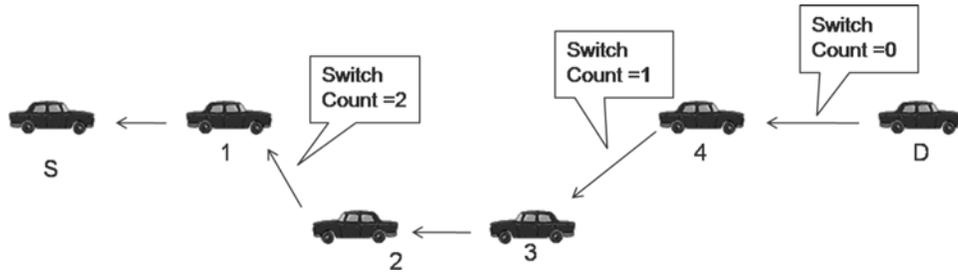


FIG. 2. SWITCH COUNT INCREMENTS IN RREP PACKET

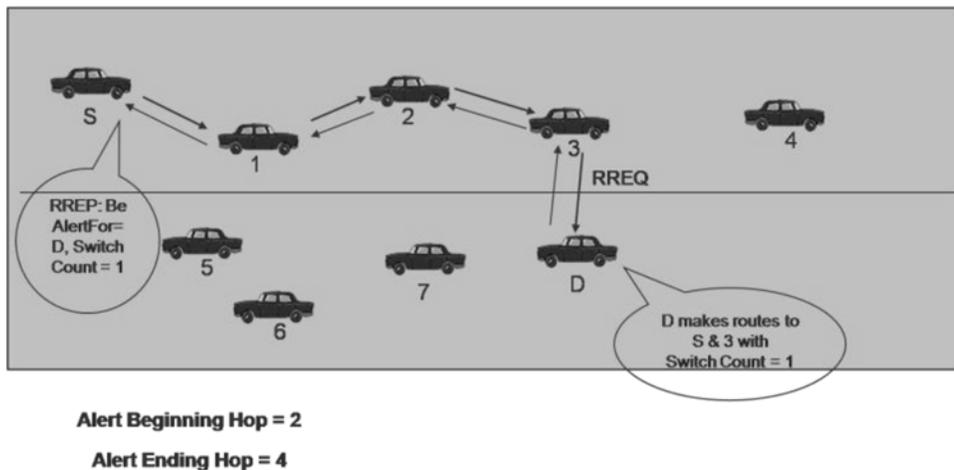


FIG. 3. DESTINATION TELLS SOME NODES TO STAY ALERT FOR IT

**Make-Before-Break:** Here the trick of Make-Before-Break comes into play. Node D in Fig. 3 puts its own IP address in the Be Alert For field of RREP. It also puts 2 and 4 in the From Hop and To Hop fields respectively. It puts 0 in Approaching Node ToDest Hops as destination itself is the Complier or Approaching node. When a node receives this RREP, it checks whether its Hop Count to D is 2 which Equation (1). Similarity, S and 1 also make entires in their west tables.

$$\text{From Hop} < \text{Hop Count} < \text{To Hop} \quad (1)$$

If it does, then it stores pertinent information in a table called Alert table. The address present in the Be Alert For field of RREP is designated as Approaching Node in the Alert table. It means that in future, upon reception of a Hello message from Approaching Node, this node has to update its routing table for D and has also to send an Update message to the Approaching node. For example, the entry made by S, 1, and 2 on reception of RREP from D in Fig. 3 is shown in Table 4. (3 does not need make entry in Alert table as its Hop Count to Node D is 1 which does not satisfy (1)). 2 makes the Alert table entry because

its Hop Count to D is 2 which satisfies Equation (1). Similarly, S and 1 also make entries in their Alert tables.

Route Update Message: Note the To Be Told field in Table 4. IP address of S is copied here from the Source IP address field of RREP. It implies that each of the nodes S, 1, or 2, on reception of a Hello message from Approaching node D, will not only change its routing table entry for D (Table 5) but will also send an Update message to D: "I am your next hop for S." On receiving this Update message, D changes its routing table entry for S as shown in Table 5. Fig. 5 illustrates the route update procedure when 2 receives a Hello message from D.

Because of the Update message, the reverse route from D to S is also maintained just like the forward route from S to D. Hence, our algorithm ensures symmetric routes.

Note that P Bit is never set unnecessarily to 1. Message is always propagated first on nodes travelling in the same direction as the forwarder node as our goal is to increase

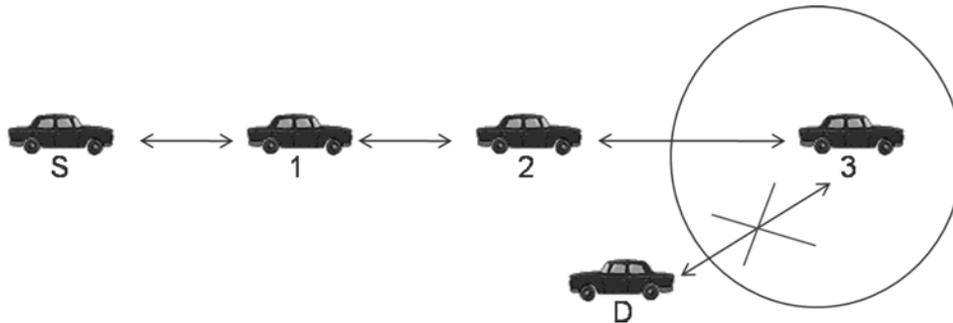


FIG. 4. MAKE-BEFORE-BREAK PROLONGS ROUTE LIFE IN SUCH SCENARIOS

TABLE 4. ALERT TABLE ENTRIES MADE BY NODES ON RECEIVING RREP

Alert Storing Nodes	To Be Told	Approaching Node	Address to be Updated	Approaching Node To Dest Hops
S, 1 and 2 (Fig. 3)	S	D	D	0
8, 9 and D (Fig. 6)	D	2	S	2



(node 2) to source node (S) (2 hops). If node D nominates node 7 (Complier 1) as Approaching node in the Be Alert For field and thus directs nodes S and 1 to perform updates, then after the second update by node S, route will have to be rediscovered as no more update is possible (Fig. 7). Node D thus nominates node 2 (Switcher 1) as Approaching node in the RREP so that nodes 8, 9, and 10 may establish connection with node 2 (3 updates) before ultimate route loss.

Table 4 shows the Alert table entries made by 8 and 9 on reception of RREP from D and D itself on reception of RREQ. Note that in the To Be Told field, address of destination node D is copied rather than that of source node S because Approaching node (the leading vehicle of a small queue which is heading towards a large queue of vehicles) here is Switcher 1, not Complier 1. In this manner, we can maintain the route between S and D for long time by incorporating as many updates as possible.

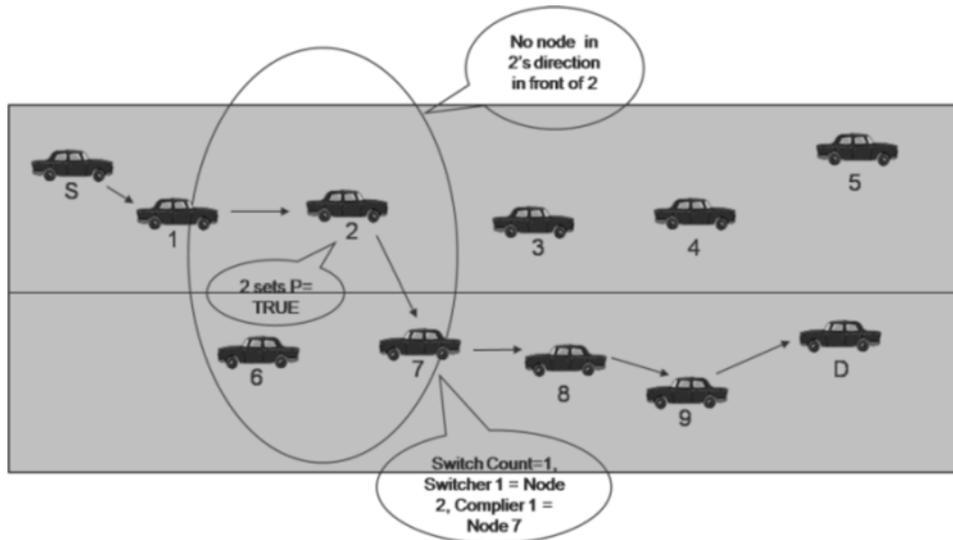


FIG. 6. LOW DENSITY, ROLE OF SWITCHER AND COMPLIER NODES

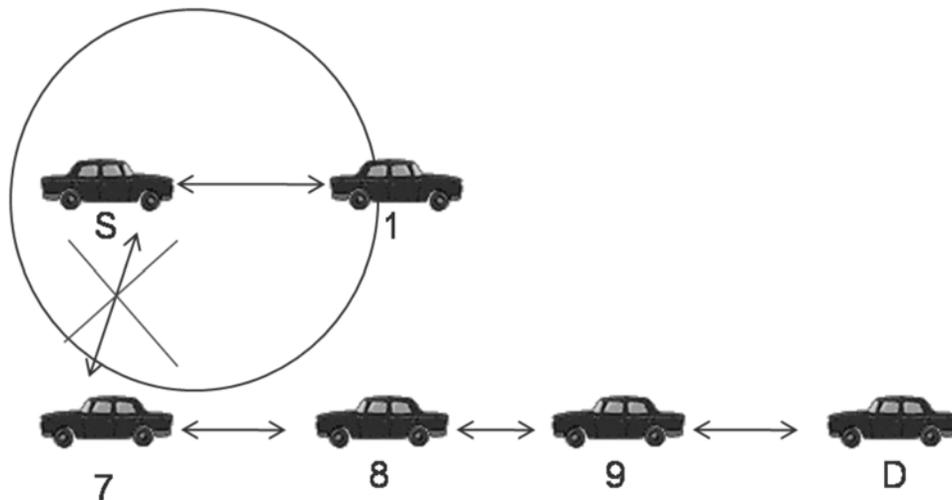


FIG. 7. IF 7 IS DECLARED AS APPROACHING NODE, ONLY TWO UPDATES (BY S AND 1) ARE POSSIBLE. 7 IS THEREFORE AN UNSUITABLE NOMINEE FOR APPROACHING NODE

**Maximum Value of Switch Count:** As stated earlier, the maximum value allowed for Switch Count is 2 in the proposed protocol. This is because the first priority in the protocol is to try the propagation of route requests through vehicles heading in same direction. Under special circumstances of non-availability of a vehicle with parallel velocity in the direction of propagation of RREQ, a node is allowed to set P bit TRUE and ask an oppositely moving node to forward RREQ. Asking this many times will kill the purpose. Also, the distance between source and destination nodes in VANETs is limited to a few kilometers and need for making this special request more than twice is not likely to arise.

In Fig. 8, we can see a real world scenario in which node S discovers route to node D. Both nodes have parallel velocities. Switch Count, beginning from 0, is incremented twice, first by node 7 (Complier 1) after setting of P = TRUE by node 2 (Switcher 1) and then by node D. Node D detects as before that nodes S, 1, and 2 are moving towards nodes 7, 8, 9, and 10. Thus it alerts nodes 8, 9, and 10 to establish connection with node 2 on reception of Hello message from it.

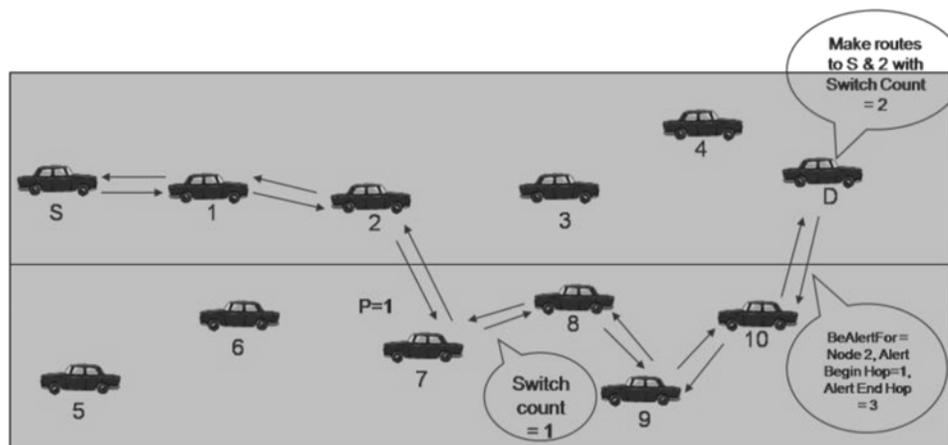
It is vital to observe that nodes 10 and D have already crossed each other at the time of route establishment

and this link will soon break which can be locally repaired by including another node between nodes 10 and D. This portion of the route breaks frequently and frequent repairs are needed. But this is only a small portion. Hence our protocol manages to keep a large portion of the route (from source node to the last intermediate node) stable.

**Local Repair:** Successive nodes having same direction can also experience link breakages between them because of having different speeds. Make-Before-Break routing has a local repair procedure for these situations. Node 1 detects in Fig. 9 that its link with node 2 has broken so it broadcasts a RREQ which reaches node 3 through node 9. As node 3 has an entry in its Alert Table which tells that D is approaching towards it, so it informs node 9 about this Approaching node D through RREP. Node 9 also becomes alert for D.

### 2.3 Node Selection Method

When RREQ is broadcast, all of the receiving nodes (albeit having same direction here) rebroadcast it. The destination responds to the first request. As pointed out by [18], successive nodes in the route may be almost radio range apart at the time of route



Node 10, Node 9 and Node 8 are Alert for Node 2

FIG. 8. A REAL WORLD LOW DENSITY CASE NODE 2 IS OBLIGED TO FORWARD RREQ TO OPPOSITELY MOVING NODE 7

establishment and may have different speeds which results in route disruption very soon. Menouar, et. al. [18] have suggested a method to tackle this problem which we use as the node selection method in our algorithm. Each node unicasts the RREQ it generates or receives to a carefully selected node which is likely to remain within its radio range for quite a while.

Consider Fig. 10. Solid cars show the current and dotted circles show the future positions of nodes *i* and *j*. Suppose the present locations of node *i* and node *j* are  $(X_{i0}, Y_{i0})$  and  $(X_{j0}, Y_{j0})$  respectively. The present distance between them is *d* and future distance is represented by *D*. Let the speeds of node *i* and node *j* be  $V_i$  and  $V_j$  respectively. If they reach the positions indicated by dotted circles in time *T*, distance between them at that time can be calculated [11]:

$$D^2 = [(X_{i0} + V_{xi} T) - (X_{j0} + V_{xj} T)]^2 + [(Y_{i0} + V_{yi} T) - (Y_{j0} + V_{yj} T)]^2 \quad (2)$$

Where  $V_{xi}$  represents the x-component of velocity of *i* and so on. Our node selection algorithm is given below:

- (1) Node *i* finds by applying Equation (2) which of its neighbors will still be in its radio range after  $T = 6$  seconds (which node will satisfy  $D < R$  where *R* is the radio range of node *i*).
- (2) Among those neighbors, the one whose *d* (present distance from node *i*) is maximum, is chosen as next hop of node *i*.

If decision for next hop is made solely on the basis of expected lifetime of the link, we may end up with smaller inter-node distances. This increases the number of hops and adversely affects overall end-to-end delay [15]. The method given above prevents this as any node which remains in the radio range of node *i* for  $T > 6$  s is a candidate for being selected as next hop of node *i*. We make the final selection on the basis of separation between selecting and to-be-selected nodes so that total number of hops does not increase significantly. Fig. 11 shows the node selection method. Except the difference in RREQ propagation, other aspects of the protocol (sending alarms and updating) remain the same.

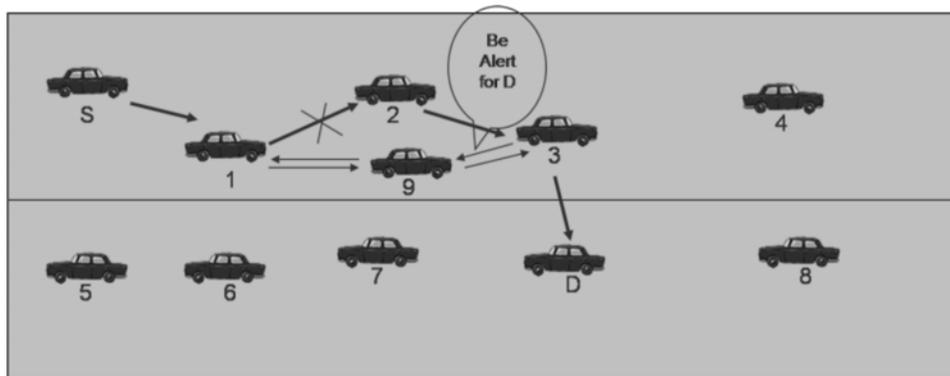


FIG. 9. LOCAL REPAIR IN PROPOSED PROTOCOL. 9, NEW IN ROUTE, IS TOLD TO REMAIN ALERT TO ESTABLISH LINK WITH D

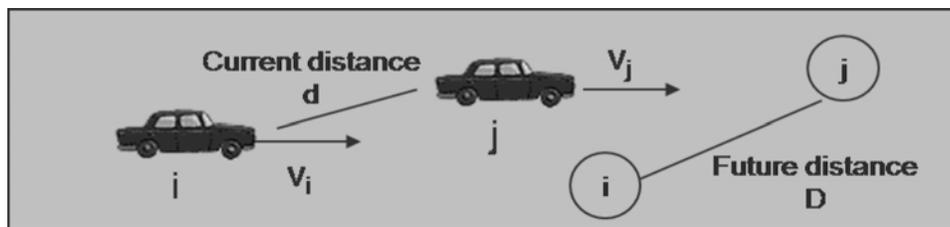


FIG. 10. NODE I DETERMINES IF J WILL BE IN ITS RANGE AFTER TIME T



We have carried out performance analysis with seven metrics as follows:

**PDR (Packet Delivery Ratio):** This is the ratio of number of packets correctly received by destination to the total number of packets sent by source. **Number of discovery RREQs:** When repair fails or is not feasible, a new RREQ is issued by source to find a new route to destination. **Route discovery time:** This means the time consumed in waiting for RREPs after issuing RREQs. **Number of repair requests:** This is the number of RREQs issued by intermediate nodes to locally find routes to destination after detecting broken links. **Route repair time:** The time

consumed by intermediate nodes to wait for RREPs after issuing RREQs for local repair. **End to end delay:** This is the average time a packet successfully delivered takes to reach destination after being relayed by source. **Route update time:** This is the total time all nodes take to establish connection with a vehicle newly arrived in their neighborhood for which they were alert beforehand. This mechanism is not present in AODV.

### 3.2 Performance Evaluation

**Scenario-1: Destination Moving Oppositely to All Other Nodes in Route (Test Case):** This scenario is depicted in Fig. 12. We have simulated this scenario for 30s. Here arises no need for P bit to be set to TRUE by any node in MBB Broadcast and no node in MBB Selection forwards RREQ to an oppositely moving node except the last forwarding to destination. Fig. 13 shows the performance metrics' variation with increasing value of maximum speed.

Fig. 13(a) plots PDR versus maximum speed. MBB Broadcast achieves higher PDR than AODV does and MBB Selection's PDR is the highest. PDR of all three tends to decrease with increasing speed however the fall in case of MBB Selection is not sharp and it manages to deliver 96% of data at maximum speed of 200 km/hr. This is because in MBB Selection, each node tries to select the next hop with similar speed. It doesn't make much

TABLE 6. SIMULATION SETTINGS TO TEST PROPOSED PROTOCOL

Parameter	Value
Dimension	3000x32m
Node Density	4 nodes in each direction every 200m
PHY Protocol	802.11a
MAC Protocol	802.11e
Date rate (Max)	6 Mbps
Range (Max)	200m
Frequency	5.9 GHz
Receive Sensitivity	-77 dBm
Data	Voice (64kbps)

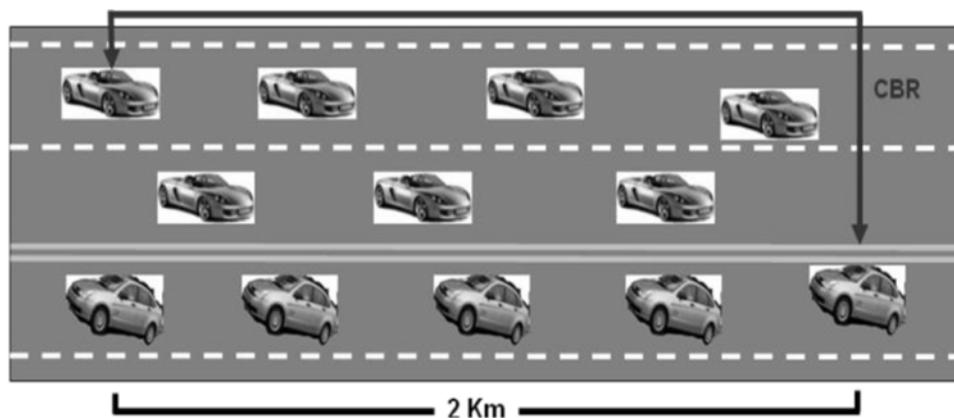


FIG. 12. TEST SCENARIO-1

difference when some nodes are travelling with very high speed when they are not selected as intermediate nodes. The reduction witnessed is due to increasing speed difference among the nodes meeting the condition given in Section 2.2. When speeds' differences are high, a node may find that none of its neighbors with different speed is expected to remain within its range for 6s and the only option available is to select a neighbor with same speed as it has, no matter how close the two nodes may be. So PDR doesn't fall significantly. At high speeds, the gap between PDR of MBB Broadcast and that of AODV narrows. The reason is the increase in frequency of link breakages between nodes having same direction at high speed variation.

Fig. 13(b) shows number of discovery route requests. With increasing speed difference, RREQs generally increase in all three protocols. Number of RREQs is the highest in AODV and is the lowest in MBB Selection. The route discovery time after making these requests is shown in Fig. 13(c). Number of RREQs and route discovery time both increase with increasing maximum speed which is expected. Interestingly number of discovery RREQs in MBB Selection in the period from 170 km/h maximum speed to 200 km/h maximum speed doesn't increase. Same is the case in the period from 110 km/h maximum speed to 140 km/h maximum speed. But we can see rapid rise in number of repair requests in these periods as evident in Fig. 13(d). Thus the total number of control messages continues to increase. (A RREQ has to be initiated when the link between source node and first intermediate node breaks). Fig. 13(g) shows end to end delay in all three protocols increases with increasing maximum speed. This is a direct follow-up of PDR trend. In MBB, PDR is high so delay is low.

Fig. 13(d) shows number of route requests issued for local repair and Fig. 13(e) shows the corresponding time spent to wait for replies to local repair requests. We observe that more repairs are possible in MBB Broadcast than in AODV. This is understandable because in AODV,

a node often finds that it cannot repair a route locally as there is danger of loop formation if RREQ reaches the source node. In contrast, MBB is direction-aware routing where RREQ for repair is sent in the direction of destination node only thwarting all chances of loop formation. Repairs are less in MBB Selection because need for repairing route seldom arises thanks to stable routes (leading to high PDR).

Fig. 13(f) plots route update time versus maximum speed. The time spent is very small i.e. a few hundred milliseconds which indicates the success of our protocol. If source and destination nodes are 2 km apart in the beginning and radio range of each node is 200 m, then there are approximately 10 or more hops in the route. Total update time of 200 milliseconds means one update is accomplished in about 20 milliseconds. On reception of Hello from the Approaching Node, the receiving node waits randomly for a few milliseconds and then transmits Update message to it. Therefore we see randomness in total update time and it does not increase or decrease with increasing maximum speed. As two nodes take very small time to establish connection, they have much time available to exchange data while they are still in each other's range.

**Scenario-2: Low Density with Source and Destination Moving in Same Direction (Real World Case):** The snapshot of this scenario is given in Fig. 14. Simulation time for this scenario is 15 seconds. The time is kept less to keep the vehicles within the Qualnet window. There is no vehicle moving towards right in a 200 m long portion of road. When MBB protocol is run in this scenario, a vehicle on the edge of this depleted region has to assume the role of Switcher 1 (through setting P bit to TRUE in MBB Broadcast and unicasting RREQ to an oppositely moving node in MBB Selection) and Switch Count is already 1 when RREQ reaches destination. The destination makes routing table entry for source with Switch Count = 2 because it is moving away from the previous node.

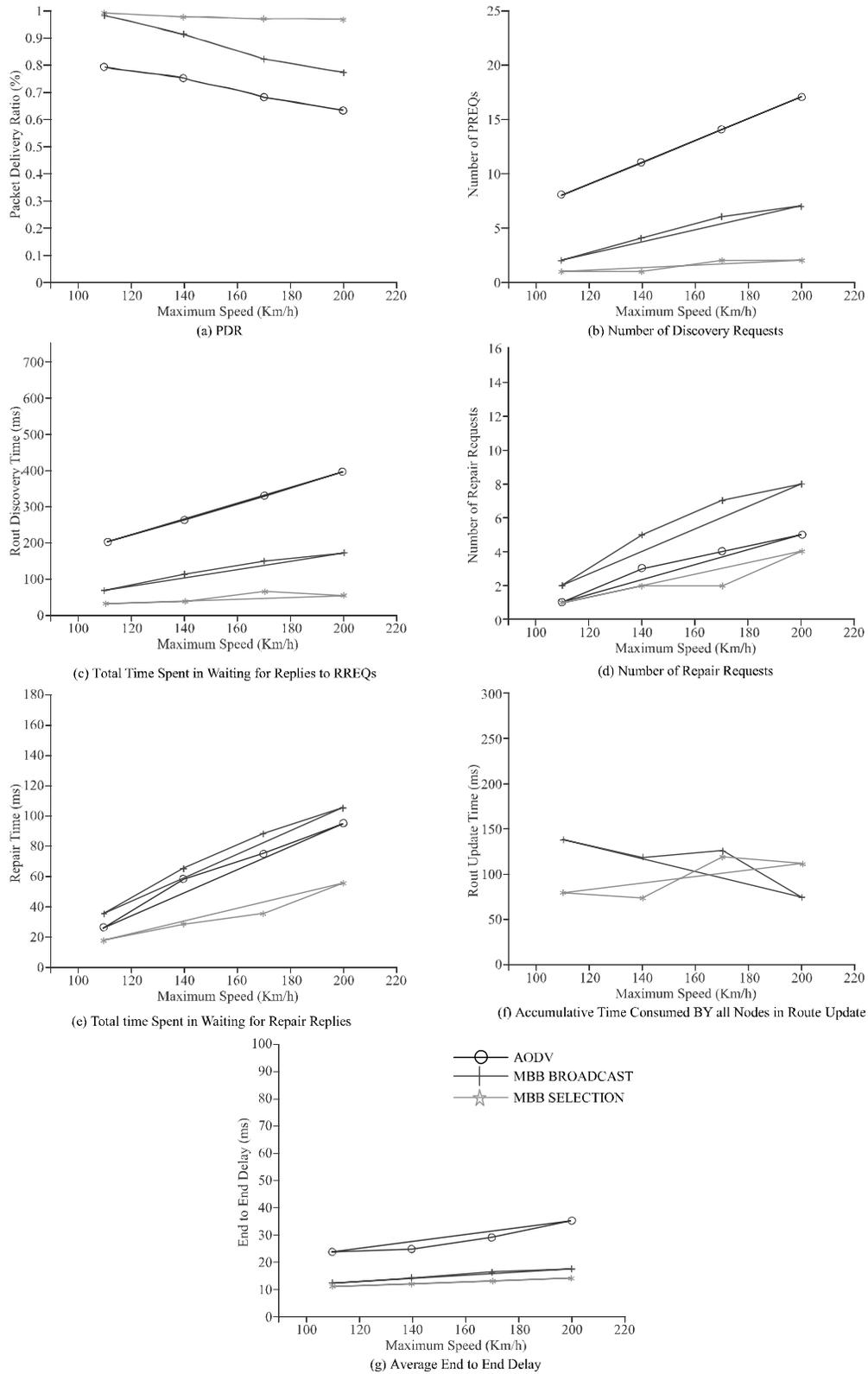


FIG. 13. RESULTS OBTAINED FOR SCENARIO-1

Fig. 15 shows the performance metrics' behavior with increasing maximum speed. Fig. 15(a) reveals that PDR of MBB Selection is the highest once again. The PDR for all three protocols however remains lower than that in the previous scenario. As destination is moving away from the last intermediate node, the sub-route between the destination and the previous node breaks more frequently. Fig. 15(b) shows that the number of discovery RREQs generally increases with increasing maximum speed. In the periods from 110-140 km/h and from 170-200 km/h, the number of initiated requests for discovery for MBB Selection doesn't appear to be increasing but the total number of control messages (requests for discovery plus requests for repair) is increasing all the time as obvious through Fig. 15(d). As a consequence of randomness, link may break first between source node and the next node resulting in a discovery request or between any two intermediate nodes resulting in a repair request.

Again, route update time is very small (Fig 15(f)), delay increases with increasing maximum speed (Fig. 15(g)), and route repair time remains in milliseconds for all three protocols (Fig. 15(e)). Fig. 15(d) shows that in contrast with the previous scenario, the number of requests for repair in case of MBB Selection is higher than in case of MBB Broadcast. This is understandable. Repairs are needed to maintain the last portion of the route where destination is always moving away from the previous

node. MBB Selection manages to keep the sub-route from source node to the last intermediate node intact by virtue of careful node selection and only needs to do repairs in this last portion.

#### 4. CONCLUSIONS

Vehicles can move with very high speeds in a highway environment. Partitioning of network in one direction under such conditions is not unusual. This brings in the inevitability of including some nodes moving in direction opposite to all others in the route.

Our algorithm prevents the disruption of route caused by link breakages between these oppositely moving nodes. Simulation results show that our "Make-Before-Break" routing protocol outperforms AODV, yielding packet delivery ratio 31% higher and end to end delay 20% lower than the corresponding outputs of AODV.

At night, traffic is less on highways causing frequent partitions. Our protocol can be used in such conditions. We suggest our protocol to be used in entertainment applications requiring communication between oppositely moving nodes as well. For example, if A is heading on a highway towards a tourist resort and B is traveling on the same highway, going back after enjoying trip to the same resort. B can send video clips to A epitomizing the beautiful places there.

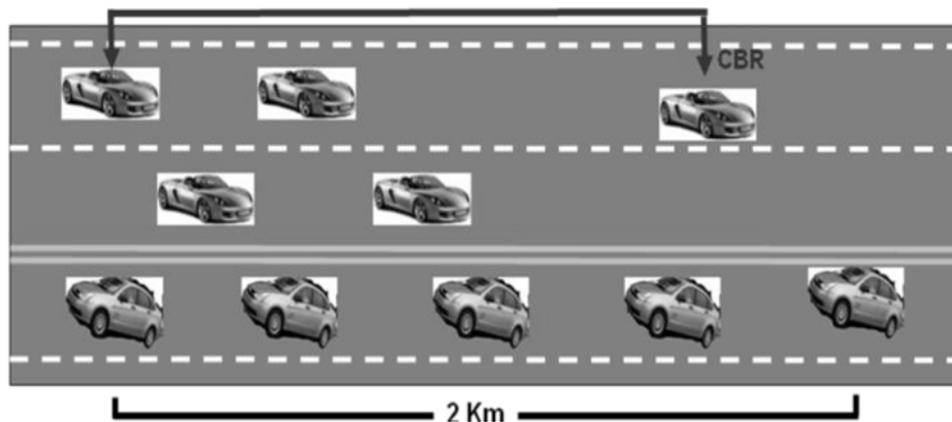


FIG. 14. REAL WORLD SCENARIO-2

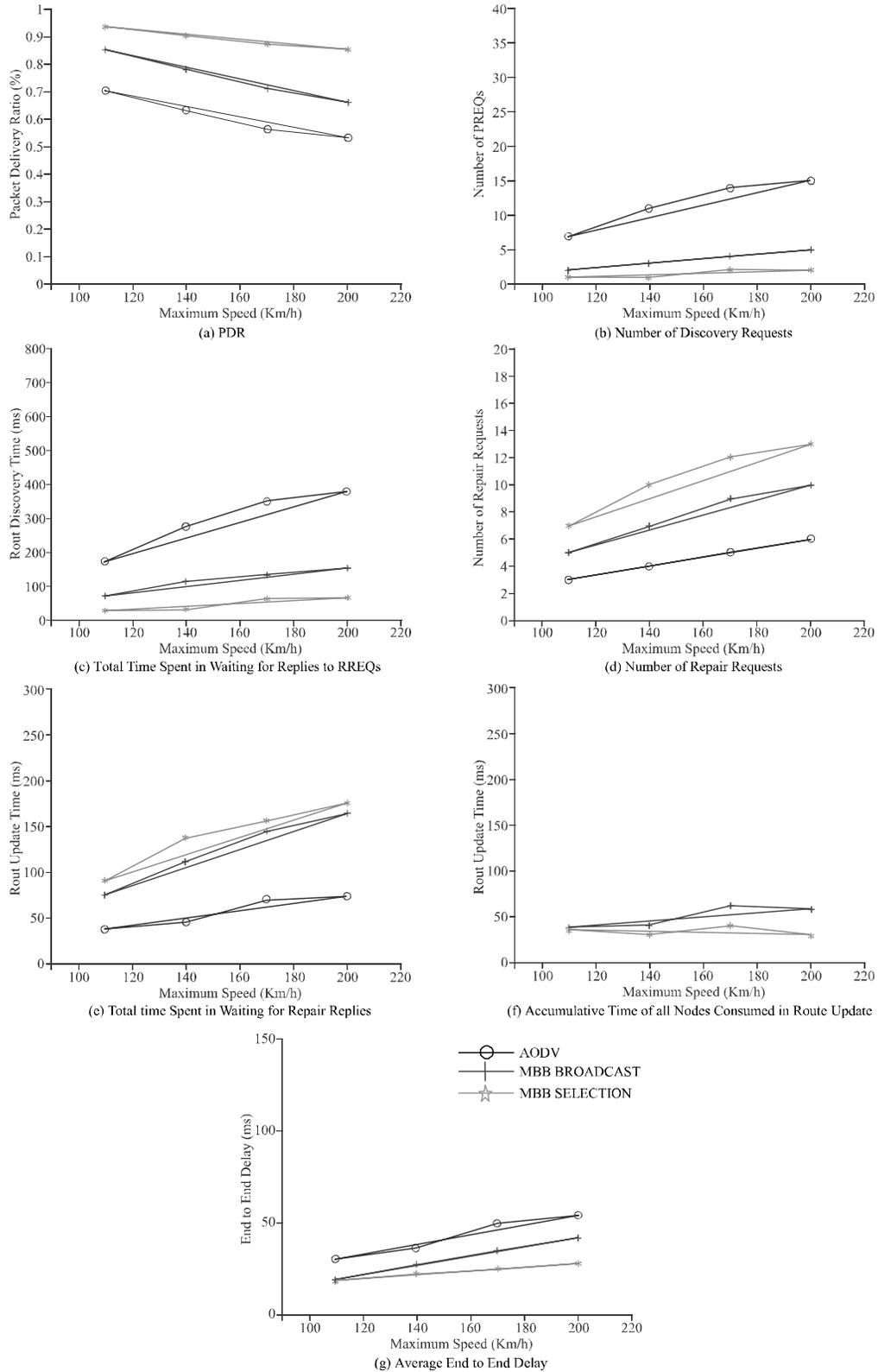


FIG. 15. RESULTS OBTAINED FOR SCENARIO-2

## ACKNOWLEDGEMENT

This work was supported in part by the South Korean Government under BK-21 program.

## REFERENCES

- [1] Liang, W., Li, Z., Zhang, H., Wang, S., and Bie, R., "Vehicular Ad Hoc Networks: Architectures, Research Issues, Methodologies, Challenges, and Trends", *International Journal of Distributed Sensor Networks*, Article ID 745303, pp. 11, December, 2014.
- [2] Karagiannis, G., Altintas, O., Ekici, E., Heijenk, G., Jarupan, B., Lin, K., and Weil, T., "Vehicular Networking: A Survey and Tutorial on Requirements, Architectures, Challenges, Standards and Solutions", *IEEE Communications Surveys & Tutorials*, Volume 13, No.4, pp. 584-616, Fourth Quarter, 2011.
- [3] Nzouonta, J., Rajgure, N., Guiling W., and Borcea, C., "VANET Routing on City Roads Using Real-Time Vehicular Traffic Information", *IEEE Transactions on Vehicular Technology*, Volume 58, No. 7, pp. 3609-3626, September, 2009.
- [4] Granelli, F., Boato, G., and Kliazovich, D., "MORA: A Movement-Based Routing Algorithm for Vehicle Ad Hoc Networks", *IEEE Workshop on Automotive Networking and Applications (AutoNet 2006)*, San Francisco, USA, December, 2006.
- [5] Li, C., Zhao, C., Zhu, L., Lin, H., and Li, J., "Geographic Routing Protocol for Vehicular Ad Hoc Networks in City Scenarios: A Proposal and Analysis", *International Journal of Communication Systems*, Volume 27, No. 12, pp. 4126-4143, 2014.
- [6] Fonseca, A., and Vazão, T., "Applicability of Position-Based Routing for VANET in Highways and Urban Environment", *Journal of Network and Computer Applications*, Volume 36, No. 3, pp. 961-973, 2013.
- [7] Sharef, B.T., Alsaqour, R.A., Ismail, M., and Bilal, S.M., "A Comparison of Various Vehicular Ad Hoc Routing Protocols Based on Communication Environments", *Proceedings of 7<sup>th</sup> International Conference on Ubiquitous Information Management & Communication*, pp. 48, January, 2013.
- [8] Dua, A., Kumar, N., and Bawa, S., "A Systematic Review on Routing Protocols for Vehicular Ad Hoc Networks", *Vehicular Communications*, Volume 1, No. 1, pp. 33-52, 2014.
- [9] Farooq, W., Khan, M.A., Rehman, S., and Saqib, N.A., "A Survey of Multicast Routing Protocols for Vehicular Ad Hoc Networks", *International Journal of Distributed Sensor Networks*, Article ID 923086, pp. 8, 2015.
- [10] Karp, B., and Kung, H.T., "GPSR: Greedy Perimeter Stateless Routing for Wireless Networks", *Proceedings of 6th Annual International Conference on Mobile Computing and Networking ACM*, pp. 243-254, August, 2000.
- [11] Lee, H., Lee, Y., Kwon, T., and Choi, Y., "Virtual Vertex Routing (VVR) for Course-Based Vehicular Ad Hoc Networks", *IEEE Conference Wireless Communications & Networking*, pp. 4405-4410, 11-15 March, 2007.
- [12] Ramachandran, L., Sukumaran, S., and Sunny, S.R., "An Intersection Based Traffic Aware Routing with Low Overhead in VANET", *International Journal of Digital Information & Wireless Communications*, Volume 3, No. 2, pp. 50-56, 2013.
- [13] Capone, A., Pizziniaco, L., Filippini, I., and de la Fuente, M.A.G., "A SiFT: An Efficient Method for Trajectory Based Forwarding", *Proceedings of 2nd International Symposium on Wireless Communication Systems*, pp. 135-139, 5-7 September, 2005.
- [14] Alotaibi, E., and Mukherjee, B., "A Survey on Routing Algorithms for Wireless Ad-Hoc and Mesh Networks", *Computer Networks*, Volume 56, No. 2, pp. 940-965, 2012.
- [15] Kumar, D., Kherani, A.A., and Altman, E., "Route Lifetime based Interactive Routing in Intervehicle Mobile Ad Hoc Networks", *INRIA Research Report No. RR-5691*, Sophia-Antipolis, France, 2005.
- [16] Drawil, N.M., and Basir, O., "Intervehicle-Communication-Assisted Localization", *IEEE Transactions on Intelligent Transportation Systems*, Volume 11, No. 3, pp. 678-691, September, 2010.

- [17] Abboud, K., and Zhuang, W., "Stochastic Analysis of a Single-Hop Communication Link in Vehicular Ad Hoc Networks", *IEEE Transactions on Intelligent Transportation Systems*, Volume 15, No. 5, pp. 2297-2307, October, 2014.
- [18] Menouar, H., Lenardi, M., and Filali, F., "Movement Prediction-Based Routing (MOPR) Concept for Position-Based Routing in Vehicular Networks," *IEEE 66<sup>th</sup> Conference on Vehicular Technology*, pp. 2101-2105, September, 2007.
- [19] Sahu, P.K., Wu, E.H., Sahoo, J., and Gerla, M., "BAHG: Back-Bone-Assisted Hop Greedy Routing for VANET's City Environments", *IEEE Transactions on Intelligent Transportation Systems*, Volume 14, No. 1, pp. 199-213, March, 2013.
- [20] Fülller, H., Mauve, M., Hartenstein, H., Käsemann, M., and Vollmer, D., "Mobicom Poster: Location-Based Routing for Vehicular Ad-Hoc Networks", *ACM SIGMOBILE Mobile Computing & Communications Review*, Volume 7, No. 1, pp. 47-49, 2003.
- [21] Li, F., and Wang, Y., "Routing in Vehicular Ad Hoc Networks: A Survey", *IEEE Vehicular Technology Magazine*, Volume 2, No. 2, pp. 12-22, June, 2007.
- [22] Suryaprasad, D., "Simulation-Based Performance Evaluation of AODV Routing Protocol for Ad Hoc Mobile Wireless Networks", Master's Thesis, Atlantic University, Florida, 2003.