A Movement Prediction-based Routing Protocol for Vehicle-to-Vehicle Communications

Hamid Menouar[†], Massimiliano Lenardi[†], and Fethi Filali[‡]
[†]HITACHI Europe, "Le Thélème", 1503 Route des Dolines, 06560 Sophia Antipolis, France
Phone: (+33)4 8987 4100, Fax: (+33)4 8987 4199
E-mail: {Hamid.Menouar, Massimiliano.Lenardi}@hitachi-eu.com
[‡]Institut Eurecom, 2229 route des Crêtes, BP-193, 06904 Sophia-Antipolis, France
Phone: (+33)4 9300 2916, Fax: (+33)4 9300 2627
E-mail: Filali@eurecom.fr

Abstract

Vehicle-to-Vehicle (V2V) Communications are going to be an important evolution of the current wireless communications, which are mainly based on the low mobility of nodes. Adapting the current wireless technologies to V2V applications is mandatory in order to achieve efficiency and reliability. In this work, we propose a routing framework based on the enhanced cooperation between the medium access and network layers. The target is to take into account the inherent characteristics of moving vehicles, in order to have an efficient cross-layer architecture for vehicular environments. First, we introduce an algorithm which takes into account position, direction and speed of vehicles. Then, we apply this concept to a reactive routing protocol and we include preliminary simulations results using NS-2.

1 Introduction

Enhancements in transportation technologies will consider, besides traditional aspects such as security and driving conditions, the ability of vehicles to communicate, including the connection of vehicle equipments to the Internet. However, in order to provide Internet-access capability in an efficient way, it would be needed to resolve several technical challenges, from gateways optimal placement in the roads to handover management between gateways; this is out of scope in this work.

On the other hand, several applications may be provided for Vehicle-to-Vehicle or Car-to-Car Communications (V2VC or C2CC). Indeed, vehicles can exchange real-time information, drivers can be automatically assisted, or passengers play distributed games, etc. In this paper we jointly address routing and Medium Access Control (MAC) issues for V2VC. Our target is to optimize the lower OSI layers for vehicular environments. Mobile Ad-hoc NETwork (MANET) topology-based routing protocols are not suitable (as they are) for V2VC due to the high mobility and fast topology changes [2], [3]. It's obvious that whatever the used routing protocol, if the node mobility is high applications may suffer from service interruption even in presence of strong handover mechanisms which will be in this case called more frequently.

When thinking at V2VC as a special case of MANET communications, not only the nodes are vehicles and not simple laptops or PDAs, but also we don't have constraints on power resources. Having more resources for V2VC is an important advantage, since these networks provide larger capacities (in terms of both storage and power) on the nodes, which can then have long transmission ranges and virtually unlimited lifetimes [1]. Furthermore, in vehicular networks, the nodes can be equipped with a positioning system, such as GPS, that can be used continuously, without power constraints.

Another advantage in such networks is the non-random mobility of the nodes (vehicles); generally it is limited by roads which can be represented by a digital maps. Also, the vehicle movements are limited by the road rules which again may be digitally mapped.

An efficient support of access and routing protocols in vehicular environment is then facing issues like: available bandwidth, hidden and exposed nodes, high mobility, heterogeneity, node movement, fast speed, obstacles and fast handover.

In this work, we propose a movement- and prediction-based routing protocol for vehicle wireless networks. Basically, it tries to predicts the future nodes' positions in order to avoid link ruptures so that frames loss rate is reduced while improving the network efficiency.

The remainder of this paper is organized as follows: Section 2 recalls the principles of cross-layer paradigm with regard to wireless networks. Our proposed routing algorithm is detailed in Section 3, while Section 4 describes the implementation in the reactive routing protocol AODV. Preliminary simulations and numerical results are provided in Section 5 and Section 6 concludes this paper and outlines future work.

2 Cross-Layer Paradigm in Wireless Networks

Cross-layer paradigm enables layers to request and use some measured parameters from each other and not only to send/receive their PDUs (Packet Data Unit) [4]. These parameters can be signal level, neighbor addresses, available bandwidth, etc.

The cross-layer approach could be applied among all layers even if they are not adjacent. In this paper, our focus is limited to the lower OSI layers, i.e. the physical, the access and the network layers. Some measures can not, in general, be provided by the layered approach, so a measurement module has to be developed, interfacing inter-working layers.

Some of the PHY parameters that can improve the routing efficiency are:

- Channel quality: this parameter is important for QoS routing. For example, an application that tolerates losses can be sent even when channel conditions are relatively bad.
- Position, direction and speed: we suppose that these parameters are locally provided by a positioning system.

Some of the MAC parameters that can improve the routing efficiency are:

- Neighborhood: the MAC layer receives broadcast beacons and is able to maintain and process the list of its neighbors MAC addresses. On the other hand, the two-hop neighbors information is useful to optimize the broadcasting mechanism used by the routing protocol for control purposes.
- Speed and direction: to improve the data routing in a vehicle network, we propose to use these information coming from intermediate node MAC layers in order to choose, among existing routes, the better one.

3 Movement Prediction-based Routing (MOPR) Algorithm

3.1 Preliminaries and short overview

Supposing that we have several potential multi-hop routes between a source vehicle and a destination vehicle, we propose to choose the route which is most stable when considering the movement conditions of the intermediate nodes with respect to the source and the destination nodes. The intermediate nodes can be either other vehicles or stationary nodes (gateways) along the roads.

This MOvement Prediction based Routing (MOPR) algorithm, by knowing speeds and directions of the nodes involved in the routes (including source and destination), can roughly predict their positions in the near future; eventually, by knowing the size of the data to send, the algorithm can know how long the transmission of each data frame will take.

Therefore, the optimal route selection for data transmission will provide the route composed by intermediate nodes that are not likely to cause a rupture of the transmission in the near future because of their mobility and that can support the transmission for enough time. This approach should help as well in minimizing the risk of broken links and in reducing data loss and link-layer and transport retransmissions.

An example illustrating the operations of MOPR is given in Figure 1, where a *node A* wants to send a data information to *node B*. Two routes are available to route these data, namely "Route 1" and "Route 2". Node A should choose one of the two available routes; if it chooses the first one (A, 1, 2, B), then, in the near future, as shown in Figure 1, the link between *node 1* and *node 2* will be cut because of their movement in opposite directions.

Our algorithm will instead use the movement information of nodes to choose the best available route; in our example the route (A, 3, 4, 5, B) is selected, so that we can avoid the link break until the end of data transmission. This decision is made despite that *node 4* has not the same direction of nodes *A* and *B*. Hence, it's possible to have intermediate nodes with different directions when this situation does not cause route break during the time needed for transmitting a given frame.

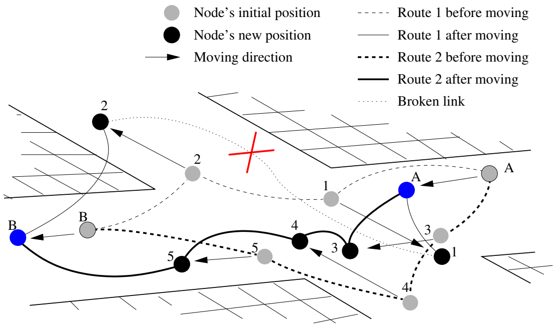

Figure 1: An example showing MOPR basic operations.

We suppose that each vehicle has a specific equipment that allows it to discover its current position, speed, and direction information, and if possible the number of the street it is running along. This last parameter could be made available by nowadays navigation systems data stored on the vehicles or by some RFID-like tags on the road broadcasting street IDs. The utility of the street ID is clear when we think at superposed roads (bridges, etc.) with (or without) different driving directions.

Figure 2 shows the diagram used for describing the direction of a node. For position encoding facility, we associate to each direction a specific number: 1 to North, 2 to East, 3 to South, and 4 to West. These numbers will be used by vehicles to precise their directions. For example, the direction of a vehicle moving toward North/East with an angle of 30 degree w.r.t. North is encoded as [1.2.30].

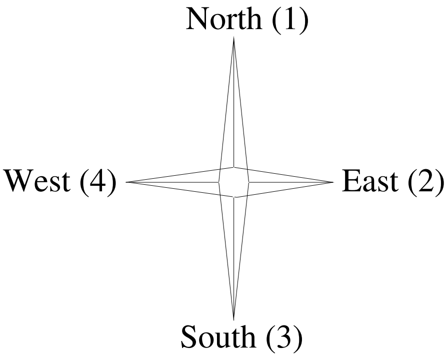

Figure 2: Direction diagram.

3.2 MOPR algorithm description

In our proposal, when a vehicle A has data to send to a vehicle B, it should first look at all available routing routes provided by a classical multi-route routing protocol. Then, among these routing routes, it selects the most stable one. A stable route is the one composed by stable nodes. The questions that we are going to answer hereafter are: *what is a stable node?* and *how to know which nodes are stable?*

A stable node is the one whose mobility (relatively to the movement of the source and the destination) will not cause broken links in the routing route during the time needed to the data transmission which depends on the size of the data to send, and on the rate of the sender and intermediate nodes.

In our contribution, a stable intermediate node should therefore move in a “similar” direction and run with a “similar” speed compared to the source and the destination vehicle directions and speeds. We propose that its speed should be in the range between the speeds of A and B, while its direction should be in between the directions of A and the one of B.

Therefore, a source node A, having multiple routes available to destination node B provided by a classical routing protocol, should proceed with the following algorithm steps in order to find the stablest route to deliver its data:

```
// R: maximum communication range
// n:  number of available routes to
destination
// m:  number of nodes in a route

// The ‘‘pos’’ function estimates the
position of a vehicle after a defined time.
// Time:  the time needed to transmit a
data.

procedure pos (id, Time);
  Position:=Position(id)+(speed_node(id)*Time);

  Return position;
end;
```

We can also improve the estimation of the future vehicle’s positions using a digital map, and eventually all involved vehicles can be also positioned on. The advantage of this proposal is that the predicted positions will be more realistic, like the one illustrated in Figure 3 where the vehicle position at t_0+t estimated without the help of a digital map is not the true one (unless an accident is happened).

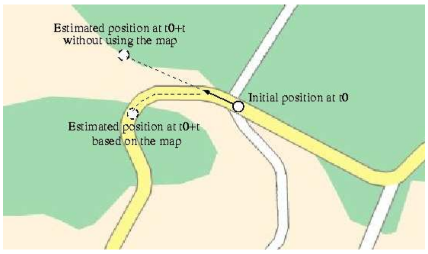

Figure 3: Position estimation based on map.

Supposing that transmission should start at t_0 and the time needed to transmit data is T, the algorithm at first estimates the position of the nodes at time $t_0 + T$ (end of transmission), then it estimates the distances at time $t_0 + T$ (taking into account a processing time) between each node and its neighbors in the route, looking if these distances are bigger than the communication radio range, which means that that route is not to be considered stable. Only if those distances are still smaller than the radio communication range that node is a stable node for that transmission.

```
// $\varepsilon$:  guard distance (processing time)
j := 0;
for k := 1 to n do
  Break := False;
  for i = 1 to m-1 do
    if Distance(pos(i-1,T),pos(i,T)) $\geq$ R-$\varepsilon$
and
    Distance(pos(i,T),pos(i+1,T)) $\geq$ R-$\varepsilon$
then
      Break := True;
    end if;
  end;
  if Break = False then
    Select[j] := k;
    j := j + 1;
  end if;
end;
```

The above procedure is executed for each available route provided by a multi-route routing protocol, and, if at the end of the procedure the *Break* value is *False* (i.e. all the intermediate links in this route will not be cut during the time (T) needed to send all the packets), then the concerned route (Route k) will be selected and added to *Select* table.

The following code is then executed in order to select among the routes saved in *Select* table the one where most of the intermediate nodes are moving in a "similar" direction and with a "similar" speed compared to both source A and destination B.

```
// A: source vehicle
// B: destination vehicle
// "Dir" function gives a node direction
// "speed" function gives a node speed

Select_Route := Select[0];
Better := m;
for k = 0 to j do
  Nbr := 0;
  for i = 1 to m-1 do
    if speed(A) ≥ speed(B) then
      if (Dir(i) ∉ range(Dir(A),Dir(B))) or
      (speed(i) ≤ speed(B) and
      (speed(i) ≥ speed (A)) then
        Nbr := Nbr + 1;
      end if;
    else
      if (Dir(i) ∉ range(Dir(A),Dir(B))) or
      (speed(i) ≥ speed(B) and
      (speed(i) ≤ speed (A)) then
        Nbr := Nbr + 1;
      end if;
    end if;
  end;
  if Nbr < Better then
    Select_Route := Select[k];
    Better := Nbr;
  end if;
end;
```

At the end of the above procedure *Select_ Route* will be the best available routing from source A to destination B in terms nodes' movement and transmission time.

Above we mentioned the possibility to use the number of the street associated to the running node in order to improve movement prediction. In this case, the route selection process could be better, for example by taking into account the number of intermediate vehicles driving on the same road of the source and/or the destination. Another possibility is to take into account the 2-dimensional scenarios where obstacles and buildings affect the routing strategy, like proposed in [3] where authors use the navigation system for optimizing the radio coverage and consequently the routing.

4 A "Reactive" Implementation of MOPR

Previously, we said that a multi-route routing protocol is needed to provide all available routes between the source and the destination vehicles, among which our algorithm will then choose the best one to use.

In this section we apply the algorithm above to the classical reactive routing protocol AODV [5]. Our proposal can further improve the quality of this protocol when used for vehicular networks and V2VC.

Generally speaking, AODV will build the route to the destination adding one intermediate node after the other, by applying the "Route Request (RREQ) - Route Reply (RREP)" procedure. The source broadcasts labeled RREQs and when the destination is reached by one of the RREQ packets it replies via the route constructed by the RREQ (each intermediate node adds in its routing table the node ID from where the RREQ came), confirming the route itself as the chosen one. If the destination gets multiple RREQs with the same label, AODV will chose the route with the smaller number of intermediate nodes (hops). An extension to the basic AODV scheme is the one of maintaining multiple routes as proposed in [6] and [7].

Because of the particular scenarios of our work (roads), the nodes are moving following specific directions and are constrained by limited lateral movements, so we slightly modified the basic AODV operation so that an intermediate node who receives multiple RREQs with same source and same route ID (and possibly with same previous-hop) will check if all previous hops in the received RREQs are identical before discarding any of them. Of course we include as well an anti-loop check in those modifications. Doing this way, the destination vehicle will possibly receive multiple RREQs, which can even come from the same previous hop, and it will then send RREPs to all of them. This process may increase the used signaling bandwidth (see section 5.2), but the benefits for short- or medium messages transmission (like warnings) will then come from the better route selection procedure. Additionally, we avoid recurrent link failures occurring when using the basic AODV which generates lot of RERR/RREQ/RREP AODV messages to look for a new path to the destination.

Our idea is to integrate MOPR concepts in the route selection; the algorithm will chose the best route by using other criteria than hop counters. Similar attitudes can be found in [8, 9, 10] where authors modify AODV for energy saving purposes (power-aware routing). A work on Geographical Routing extending AODV has been proposed as well, see [11], but in that work, authors focus on the multicast aspect of vehicular AODV routing. In [12] the use of a cluster-based location routing is proposed; there is an initial cluster formation phase, followed by a Location Discovery phase to get the position of the destination before starting the data transmission. Interesting is the introduction of a Microscopic Traffic model for simulating “fleets” around the cluster headers. Concerning mobility modeling, in [13] the authors proposed a framework for mobility model generation for V2VC. These works can turn to be useful for improvements of our algorithm.

We suppose, as shown in Figure 4, that for every entry in the routing table each node adds some more information: Position, Direction, Speed and possibly Street or Navigation data. So the source vehicle will decide which route to use depending on the movement information contained in the RREPs received back from the RREQ procedure and on the time needed to transmit the information. This procedure tries indeed to predict the final position of the nodes involved in the future transmission in order to prevent link failures. It was shown in [14], even if within uniformly random mobility patterns, that improving and using location prediction leads to better routing.

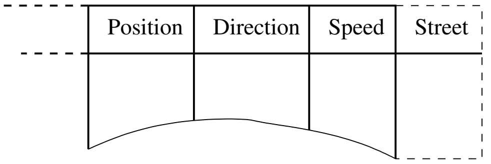

Figure 4: Routing table extension.

A simple example of our proposal process is depicted in Figure 5, where a node "A" wants to send some data to node "B". Initially, "A" broadcasts a request message to all its neighbors in order to reach the destination node "B". Each intermediate node that receives that route request message, like AODV, adds in its routing table the RREQ previous hop. Then, as shown in Figure 5, "B" receives several request messages, each one proposing a possible route from "A".

Unlike AODV, “B” will reply to all of them and each intermediate node receiving the RREP will fits in it its identifier and all its moving information. Doing that, “A” will be able to calculate the “stability” of each received route proposal using MOPR and the moving information of all intermediate nodes contained in the RREPs, and to chose the best MOPR route to “B”. In the example shown in Figure 5, [A.0.2.5.B] is selected as the best first route just because (let’s suppose) its RREQ is the first to arrive at “B”. But "A" receives later a RREP with a better route to “B”, say [A.1.4.B], so “A” will change its routing to the newer route on-the-fly, while transmitting data to “B”.

What has been presented above is an initial phase for the development of a “cross-layer” routing architecture; it is part of our future work to optimize it and to perform more comparisons by simulations (preliminary numerical results are presented in Section 5); finally, our target would be to have working prototypes.

An alternative approach would be, for example, to design a joint MAC/Routing scheme, meaning that the routing is done at the MAC layer. More precisely, the routing will not be based on the IP destination address as the case of IP routing but on a link layer label. Similar to MPLS [15], a trivial advantage of this scheme is the fast switching given that packets do not go through the IP layer to be sent to the destination. In [16], the authors propose a label-switching scheme for multi-hop wireless networks. We believe that a similar approach can be efficient for V2VC because there is a need for fast switching due to the high topology change frequency. This approach will be a subject of further investigations.

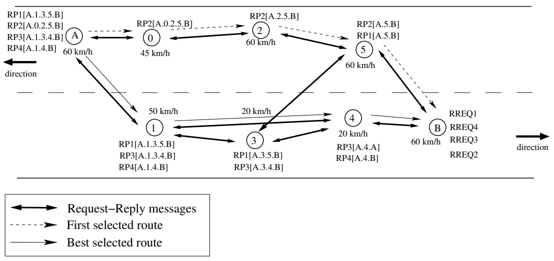

Figure 5: Simple example showing the integration of MOPR in a reactive routing protocol.

5 Simulation results

5.1 Simulation environment and scenario description

To evaluate the performance of our MOPR algorithm, we compare the simulation results of the basic AODV and of the combination of our modified AODV plus MOPR which has been implemented in NS2 v2.28 network simulator [17]. We term our algorithm AODV-MOPR.

As an example, when 120 nodes (vehicles) are simulated, Figure 6 shows the network topology. Nodes were placed over 3 crossing ways each with 4 parallel roads with alternating driving directions. Hence there are 10 vehicles per road with initial position randomly chosen along the road. The horizontal way was 2500 m long and 400 m large (100 m per road), while the vertical ones have the same width, but they are 1500 m long. When other vehicles are added to the simulations, they are placed always along the main horizontal way; for example, when 160 nodes are simulated, 80 are running along the main horizontal way (see Figure 8).

Radio propagation range was set to 250 m and channel capacity to 1 Mb/s. The vehicle mobility was constrained along the roads with a fixed direction and fixed speed randomly chosen within ranges of 30 km/h starting from [30,60] km/h until [150,180] km/h (see Figure 7).

We used the classical 802.11 Medium Access Control (MAC) functionalities, i.e. Distributed Coordination Function (DCF), Carrier Sense Multiple Access with acknowledgements (CSMA/CA with ACK) and Request-To-Send Clear-To-Send (RTS/CTS), and fragmentation, even if we suppose the messages are enough small. Traffic type was CBR, and the only transmitting source and destination were selected randomly along the horizontal road in Figure 6.

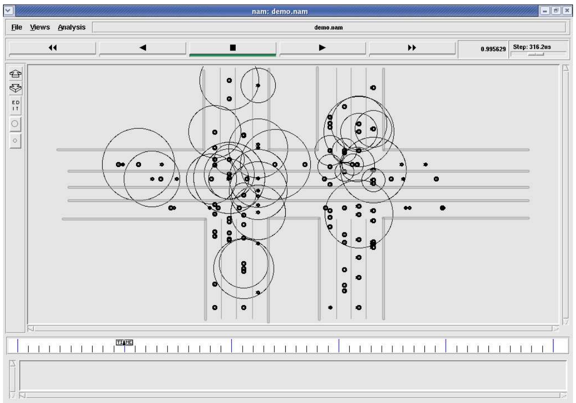

Figure 6: A screenshot showing the vehicles' distribution in the simulated region

5.2 Results and Analysis

Further investigation and simulations are needed to prove the performance of AODV-MOPR, but Figure 7 shows the percentage of link failures detected for AODV-MOPR and for the basic AODV over 1000 runs.

We set the time T needed to transfer some data from the source to the destination change to 5 seconds. Note that this parameter T does not correspond to a fixed data size to be sent, but to the period between the time when the source starts the transmission until the moment the destination gets all data which can be formed by many small packets. This means that if two vehicles have 100 km/h as difference in speed, the distance between them after 5 seconds will augment of around 139 meters, which is still less than the radio coverage of a node and so they are still reachable from each other.

We can notice here how AODV-MOPR is always detecting less link failures than basic AODV; in general 3% to 5% less link failures. The trend, anyway, seems to be that the MOPR gain is almost constant with respect to classic AODV, which selects the shortest available route. MOPR normally chooses the most stable route immediately, but this route can of course fail if the stability of the available routes is not sufficient.

In the second simulation the topology is the same, but we fixed the vehicle speed range between 60Km/h and 90Km/h. What is varying is the number of vehicles driving on the main way, the horizontal one in the figure 6: initially there are 40 vehicles, and at each time we add 20 vehicles until we reach 120 vehicles,

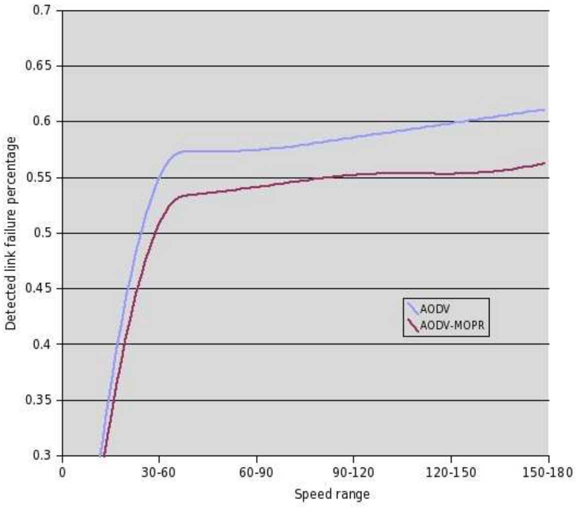

Figure 7: Percentage of detected link failures for AODV and AODV-MOPR when changing the speed range of vehicles.

i.e. 220 vehicles on all network. The results (percentage of detected link failure) obtained over 1000 runs for each point are presented in the figure 8. We notice that the gain of MOPR is again almost stable and around 7% with respect to AODV. The density of vehicles in the network plays here an important role in decreasing the number of link failures.

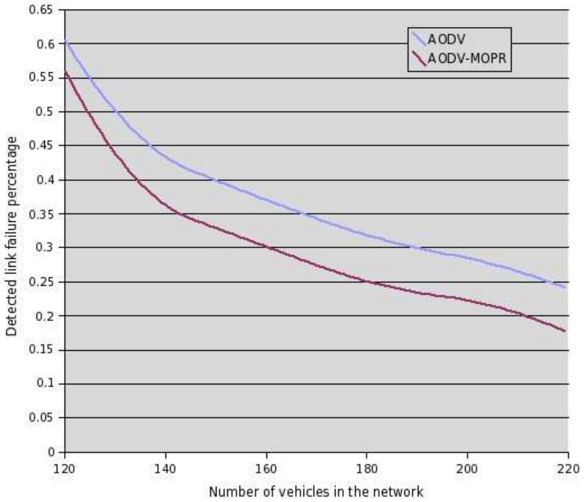

Figure 8: Percentage of detected link failures for AODV and AODV-MOPR when changing the number of vehicles in the network.

A first analysis of the increased bandwidth used by MOPR with respect to AODV shows that around 20% more bandwidth is used to manage MOPR routing, due to multiple RREPs sent by the destination node, each one increasing its size at intermediate nodes. This figure must be confirmed by other extensive simulations, but authors think that it can be even smaller because AODV must restart the RREQ procedure when a link failure is detected, and this is not taken into account in the analysis (MOPR avoids this procedure, indeed).

6 Conclusion and Future works

Because of the fast moving characteristics of vehicles and the difficulty to predict the traffic variations, it is very hard to efficiently cope with these problems while deploying methods for data routing in vehicular networks.

In this paper, we presented a part of our work, that focused on designing an algorithm that allows routing protocols to avoid links potentially broken by the node mobility during data transmission, therefore to avoid data loss and network overload caused by retransmissions. Basically, the proposed algorithm uses the moving information of vehicles to choose the best routing route. Furthermore, we applied our ideas to one of the classical on-demand reactive routing protocol, AODV.

Future works include the development of a complete cross-layer architecture (and possibly a joint MAC/Routing architecture as well) including not only information about vehicles speed and direction but also channel quality. Also, in the future we will implement our algorithm within other existing routing protocols. Some more simulations and comparisons should be provided too.

References

[1] Jun Luo and Jean-Pierre Hubaux, “A Survey of Inter-Vehicle Communication”, School of Computer and Communication Sciences, EPFL, CH-1015 Lausanne, Switzerland, Technical Report IC/2004/24, 2004.

[2] Jeremy J. Blum, Azim Eskandarian, and Lance J. Hoffiman, “Challenges of Inter-vehicle Ad Hoc Networks”, in the IEEE Transaction on Intelligent Transportation Systems, Vol. 5, No. 4, December 2004.

[3] C. Lochert, H. Hartenstein, J. Tian, H. Fussler, D. Hermann, M. Mauve, “A Routing Strategy for Vehicular Ad Hoc Networks in City Environments”, 58th IEEE Semiannual Vehicular Technology Conference VTC 2003-Fall, pp. 156-161, Orlando, FL, USA, October 2003.

[4] Liang Qin and Thomas Kunz, “Survey on Mobile Ad Hoc Network Routing Protocols and Cross-Layer Design”, Carleton University, Systems and Computer Engineering, Technical Report SCE-04-14, August 2004.

[5] Charles E. Perkins and Elizabeth M. Royer. “Ad hoc On-Demand Distance Vector Routing”, in the Proceedings of the 2nd IEEE Workshop on Mobile Computing Systems and Applications, pp. 90-100, New Orleans, LA, February 1999.

[6] Sung-Ju Lee and Mario Gerla, “AODV-BR: Backup Routing in Ad hoc Networks" Proceedings of IEEE WCNC 2000, Chicago, IL, September 2000.

[7] M. K. Marina and S. R. Das, “On-demand Multipath Distance Vector Routing for Ad Hoc Networks” , in Proceedings of the International Conference for Network Protocols (ICNP) , Riverside, Nov. 2001.

[8] N. Gupta and S. R. Das, “Energy Aware On-Demand Routing for Mobile Ad Hoc Networks”, Springer Lecture Notes in Computer Science: LNCS 2571: Proceedings of the 4th International Workshop on Distributed Computing (IWDC 2002), Kolkata, India, December 2002, pages 164-173.

[9] G. Chao, “Energy Sensitive Routing in Ad Hoc Networks”, 2nd International Workshop on Embedded Systems, Internet Programming and Industrial IT, 20-22 August, 2003,Kiel, Germany.

[10] G. Chao and R. Jantti, “A Reactive Power-Aware Routing Protocol for Ad hoc Networks”, IEEE Vehicular Technology Conference VTC2004’Spring, 17-20, May, 2004, Milan, Italy.

[11] C. Schwingenschloegl and T. Kosch, “Geocast Enhancements for AODV in Vehicular Networks”, ACM Mobile Computing and Communications Review July 2002.

[12] R. A. Santos, R. M. Edwards and N. L. Seed, “Inter Vehicular Data Exchange Between Fast Moving Road Traffic Using an Ad-hoc Cluster-Based Location Routing Algorithm and 802.11b Direct Sequence Spread Spectrum Radio”, PGNet 2003.

[13] J. Haerri, F. Filali and C. Bonnet, “A framework for mobility models generation and its application to inter-vehicular networks”, MobiWac 2005, 3rd IEEE International Workshop on Mobility Management and Wireless Access, June 13-16, 2005, Maui, Hawaii, U.S.A.

[14] S. F. Bush and N. Smith, “The Limits of Motion Prediction Support for Ad hoc Wireless Network Performance”, International Conference on Wireless Networks (ICWN-05) Monte Carlo Resort, Las Vegas, Nevada, USA, June 27-30, 2005.

[15] MPLS Resource Center, http://www.mplsrc.com

[16] A. Acharya, A. Misra, and S. Bansal, “A label-switching packet forwarding architecture for multi-hop wireless LANs”, in the Proc. of the Fifth ACM International Workshop on Wireless Mobile Multimedia (WoWMoM 2002), September 2002.

[17] http://www.isi.edu/nsnam/ns/