Virtual Virtual Circuits: One Step Beyond Virtual Mobile Nodes in Vehicular Ad-hoc Networks

Jack Fernando Bravo-Torres
Exact Sciences Area, Salesian Polytechnic University
Calle Vieja 12-30 y Elia Liut, Cuenca, Ecuador
Email: jbravo@ups.edu.ec

Martín López-Nores, Yolanda Blanco-Fernández
and José Juan Pazos-Arias
Department of Telematics Engineering, University of Vigo
EE Telecomunicación, Campus Universitario s/n, 36310 Vigo, Spain
Email: {mlnores,yolanda,jose}@det.uvigo.es

***Abstract*—*Virtual mobile nodes* (VMNs) were recently introduced as a means to address the challenges raised by the mobility of the nodes in ad hoc networks. This concept can be readily applied in the specific realm of VANETs, taking advantage of the facts that the movements of the vehicles are bound to the road and street maps, and that the variation of vehicle density has statistical behavior in time and space. Several studies have proved that VMNs can facilitate the planning of communications among the vehicles that travel through a given area. We introduce the concept of *virtual virtual circuits* (VVCs) as a means to reduce even further the number of control packets flowing through the VANETs and the computational load on the communication devices. Early results are presented that evidence significant improvements in terms of delay, packet losses and throughput.**

I. Introduction

Communication and entertainment devices are an integrated part of today's automobiles, with connectivity provided by cellular and satellite networks. *Vehicular ad hoc networks* (VANETs) have long promised to offer new communication services at much lower cost, such as the dissemination of real-time traffic information, calling and messaging between users on the road, Internet access through public gateways, transmission of *on-board diagnostics* (OBD) information, etc. However, the deployment of communication services in VANETs is still noticeable for its absence, due to the technical challenges raised by the mobility of the vehicles. This problem has been the focus of much research in the broader area of *mobile ad hoc networks* (MANETs), leading to solutions of very different natures (see [1]).

One recent idea is that of implementing a *virtualization* layer that allows the communication protocols to consider stable virtual nodes rather than the unpredictable real ones [2]. This approach is based on the notion of *virtual mobile nodes* (VMNs), which are abstract nodes that move in a predetermined manner so that each one covers a certain area. The VMNs are collectively supported by the real nodes that travel near their respective points of location, at least for some time. Yet, the direction of movement of the VMNs is not determined by the direction of movement of the real nodes. This way, much of the uncertainty of the real mobile nodes in the ad hoc wireless environment can be masked to simplify the design and operation of communication services.

The benefits of having VMNs have already been demonstrated in MANET scenarios. For example, Wu et al. [3] presented a study on reactive routing with the VNLayer virtualization system [4], comparing the AODV algorithm with a variant involving virtual mobile nodes, called VNAODV. The simulation results showed that VNAODV could deliver more packets, generate less routing traffic and create more stable routes than AODV in dense MANETs with high node motion rates. These results should be reinforced in VANET scenarios, due to the fact that the movements of the real nodes —the vehicles— are not at all random, as they are usually assumed to be in MANETs. Furthermore, the variation of vehicle density has statistical behavior in time and space, and there exist traffic models for many cities that capture the major flows of traffic, peak hours and the like. All this information, undoubtedly, facilitates the operation of the VMNs.

Nevertheless, it is possible to achieve further benefits by exploiting information not only about road traffic, but also about the origins and destinations of the communications. Bearing this in mind, we have introduced the notion of *virtual virtual circuit* (VVC), which we describe in Section II. We present the results of our first experimental results with VVCs in Section III, considering results with VMNs alone as a reference. Conclusions are given in Section IV.

II. Virtual Virtual Circuits

Traditional *virtual circuits* were introduced back in the 1970s, during the early development of communication networks, to reduce the overhead due to running the routing algorithms in each one of the nodes traversed by the data packets. The idea was to define logical paths ensuring that packets would be always delivered along the same sequence of nodes, which were therefore freed from routing decisions. Focusing on the realm of vehicular ad hoc networks, we define a *virtual virtual circuit* as a route over a sequence of VMNs, to deliver packets between its endpoints. In other words, a VVC is a virtual circuit defined over virtual nodes. There may be cases in which one of the endpoints of an VVC is fixed (e.g. a VVC may end in the area covered by a public Wi-Fi hotspot to provide Internet access to a vehicle, or take packets with OBD data to an automobile repair shop for purposes of continuous monitoring). In general, however, the endpoints of VVCs will be vehicles on the move.

Moving endpoints mean that the sequence of VMNs traversed by a VVC must evolve over time. This is easy to achieve once we can rely on the VMNs (to a greater or lesser extent) as stable repositories of information, regardless of which vehicles are currently within the area they cover. To illustrate the simplest case, suppose that vehicle $\mathcal{V}_1$ is within the area covered by $\mathcal{VMN}_r$ and it wants to exchange messages with vehicle $\mathcal{V}_2$, that is within the area covered by a neighboring VMN, $\mathcal{VMN}_s$. A VVC can then be established as the sequence $\mathcal{VMN}_r - \mathcal{VMN}_s$. If $\mathcal{V}_2$ moves on to the area covered by $\mathcal{VMN}_t$, the VVC is updated to become $\mathcal{VMN}_r - \mathcal{VMN}_s - \mathcal{VMN}_t$. If $\mathcal{V}_1$ moves the other way to the area covered by $\mathcal{VMN}_q$, the VVC becomes $\mathcal{VMN}_q - \mathcal{VMN}_r - \mathcal{VMN}_s - \mathcal{VMN}_t$, and so on.

The aforementioned changes can be accomplished just by changing the information stored in each VMN, which is basically one table linking each VVC identifier with the identifiers of the preceding and the subsequent VMNs in the sequence. This approach naturally prevents the formation of loops —and, thereby, also prevents unnecesary hops in transmitting the data packets— if, for example, one of the vehicles passes by one place twice or more times.

Clearly, the quality of service supported by a VVC is a function of the QoS supported by the VMNs it traverses, which is ultimately a function of the traffic flows and densities expected throughout the areas they cover. Just like in the use of traditional virtual circuits, it should be possible to define resource allocations and admission control mechanisms, but we have not yet made experiments with that. In what follows, therefore, we assume a best-effort policy.

III. Experimental Results

In order to assess the advantages of using VVCs in VANET communications, we have made experiments with a simulation environment that combines the well-known *ns-2* network simulator with SUMO, a road traffic simulator that models each vehicle explicitly, with its own location and speed. Following ideas already put forward in [5], we use SUMO to feed *ns-2* with realistic mobility traces; likewise, we have extended the design of the virtualization layer of [3] to manage VVCs as explained in the preceding section.

We simulated scenarios with vehicles moving over a square urban plan of 2400 m by 2400 m, the streets provided automatically by SUMO. All the street stretches had double lanes and a speed limit of 15 m/s (54 km/h). We placed one VMN at each crossroads, and then covered the length of the stretches between those junctions with as many equispaced VMNs as needed. The locations of the VMNs were fixed, but they were allowed to move whenever the area they should cover by default were likely to become empty. In such conditions, in an attempt not to lose its state, a VMN could move as if it were dragged by the last vehicle leaving the default area; if that vehicle then came across another one going in the opposite direction, the whole state of the VMN could be transmitted onto the latter so as to return to the default position.

As regards the communications, we had a variable number of users —chosen at random— initiating VoIP calls to have 1-minute to 3-minute conversations with contacts over the Internet. This type of network traffic is not very sensitive to losses, but highly sensitive to round-trip delays. The packets were routed through the VANET towards a number of Wi-Fi hotspots deployed over the city. We assumed that each vehicle was equipped with a communications/navigation device that could know its geographic location and communicate with neighboring vehicles within a range of 100 m.

By comparing the results achieved with and without VVCs in each one of the simulation scenarios, we could confirm that the use of VVCs reduces the number of control packets flowing through the VANET and the computational load on the communication devices. This lower overhead implied less time spent by each packet at the VMNs it traverses, with an average saving of 12% in round-trip delay. There were also fewer packet losses (3% fewer in average), which is probably due to the fact that the queues of packets to be forwarded from each VMN were smaller in the configuration with VVCs. Consequently, we measured an overall 14% increase in the number of conversations that could be completed with reasonable quality. In real life, the lower overhead would also imply lower battery consumption in the communication devices.

IV. Conclusion

The impact of virtual mobile nodes has only been studied thus far in MANET scenarios, using them as store-and-forward entities that had to run the routing algorithms in a datagram-like fashion. Our results reveal that VANETs are a propitious environment for a different approach, since the VMNs can be made stable enough to maintain simple information about virtual circuits established through them. Our management of virtual virtual circuits can therefore unleash the possibilities of a virtualization layer to support demanding communication services that have not been possible to date in VANETs.

Acknowledgment

This work has been supported by the Consellería de Educación e Ordenación Universitaria (Xunta de Galicia) incentives file CN 2011/023 (partly financed with FEDER funds).

References

[1] M. López-Nores, J. J. Pazos-Arias, J. García-Duque, T. Blanco-Fernández, M. Ramos-Cabrer, A. Gil-Solla, R. P. Díaz-Redondo, and A. Fernández-Vilas, "KEPPAN: Knowledge exploitation for proactively-planned ad-hoc networks," *Journal of Network and Computer Applications*, vol. 32, no. 6, pp. 1194–1209, 2009.

[2] S. Dolev, S. Gilbert, N. A. Lynch, E. Schiller, A. A. Shvartsman, and J. L. Welch, "Virtual mobile nodes for mobile ad hoc networks," in *Proc. of the 18th Annual Conference on Distributed Computing (DISC)*, Amsterdam, the Netherlands, Oct. 2004, pp. 230–244.

[3] J. Wu, N. Griffeth, C. Newport, and N. Lynch, "Engineering the virtual node layer for reactive MANET routing," in *Proc. of the 10th IEEE International Symposium on Network Computing and Applications (NCA)*, Cambridge (MA), USA, Aug. 2011, pp. 131–138.

[4] M. Brown, S. Gilbert, N. Lynch, C. Newport, T. Nolte, and M. Spindel, "The virtual node layer: a programming abstraction for wireless sensor networks," *SIGBED Review*, vol. 4, pp. 7–12, Jul. 2007.

[5] C. Lochert, M. Caliskan, B. Scheuermann, A. Barthels, A. Cervantes, and M. Mauve, "Multiple simulator interlinking environment for inter vehicle communication," in *Proc. of the 2nd ACM International Workshop on Vehicular Ad Hoc Networks*, Cologne, Germany, Sep. 2005, pp. 87–88.