C-VeT an open research platform for VANETs: Evaluation of Peer to Peer Applications in Vehicular Networks

Eugenio Giordano*†, Andrea Tomatis‡, Abhishek Ghosh* Giovanni Pau*, Mario Gerla*
*Computer Science Department - University of California, Los Angeles, CA 90095, US
e-mail: {giordano|aby|gpau|gerla}@cs.ucla.edu
†DEIS, WiLab - Università di Bologna, Bologna, 40136, Italia
‡Politecnico di Torino, Electronic Department, NAVSas
e-mail: andrea.tomatis@polito.it

I. Introduction

The UCLA Vehicular Testbed [6] has been designed to provide a flexible and fully virtualized platform to test the newly designed protocols for vehicular networks. The rationale behind a campus testbed lies on the idea that a campus may be considered a good model of a small city. In particular UCLA, with its 15 acre campus, can be compared to downtown in an average US city or to a small city itself. The campus landscape contains all the features typical of a city such as high rise buildings as well as low raise ones, parks, parking structures and roads of every size. The mobility patterns inside the campus are driven by a number of activities very similar to those present in any US downtown. For instance, the campus environment features a police department, a fire department, a post office and a facility management service that continuously patrol the campus road while on duty. Similarly to what happens in a real city, everyday routine performed by the campus citizens (i.e. students, faculties, staff, and visitors) frequently involves driving, leading to urban-like mobility patterns. Students, for example go to the Registry office for their paperwork as citizens go to the city hall, etc. *C-VeT* benefits from UCLA's campus features in terms of landscape, infrastructure and mobility to provide a realistic vehicular network testbed benefitting the research community in this field. In particular, *C-VeT* exploits the facility management mobility and the campus wide wireless infrastructure to provide a fully virtualized and web-accessible platform for researchers to develop a new generation of protocols for Vehicular ad hoc Networks (VANETs).

†This work has been partially supported by: (a) Microsoft Research through its PhD Scholarship Programme; (b) the National Science Foundation with grants CNS-0520332 and CNS-0435515; (c) the UC-Discovery initiative grant N. com07-10252; (d) the Italian Ministry for research and education under the DAMASCO initiative. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the funding agencies. Corresponding author Giovanni Pau – gpau@cs.ucla.edu.

II. Test Bed Description

C-VeT features three classes of compontents as shown in figure 1. Vehicular nodes, installed on the facility-management cars, are the core of the vehicular testbed. The campus wireless infrastructure, complemented with a custom mesh network, provides Internet access and the communication backbone. Finally, a 900MHz wide area wireless network is used to access the vehicles and manage them remotely without interfering with the on going experiments.

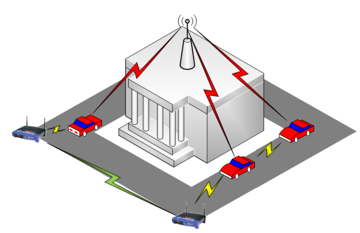

Fig. 1. *C-VeT scheme*

A. Backbone network

C-VeT's backbone is based on the UCLA's $IEEE802.11$ campus-wide wireless infrastructure and is complemented by a wireless mesh provided by MobiMesh[1](green links in fig. 1). The nodes participating in the mesh are installed in strategic campus location and perform packet routing as well as serve as access points for the mobile nodes. The mesh core network is deployed using $IEEE802.11a$ interfaces at 5.9GHz and directional antennas while the access service is offered using $IEEE802.11g$ interfaces in the 2.4GHz band (yellow links in fig. 1). The network management and monitoring infrastructure has been deployed using a wide area wireless network technology in the ISM band of 900MHz red links in fig. 1. The 900MHz digital radio, provides enough bandwidth to perform realtime monitoring of the vehicular network and maintenance of the network nodes. The 900MHz wireless infrastructure has been laid down to guarantee an independent channel to be used for network management and monitoring operations without interferences with on going experiments.

B. Mobile nodes

Each vehicle is equipped with an industrial strength Cappucino PC based on the Intel Dual Core Duo architecture. In addition, each node is equipped with:

- **GPS receiver**: which is in charge of evaluating Position, Velocity and Time (PVT) using the signal transmitted by the GPS constellation.
- **802.11 b/g MIMO wireless card**: which enables the node to communicate with the other nodes in ad hoc mode.
- **Radio modem**: which is in charge of managing the connection with the Backbone network using the 900MHz ISM band.

Each Cappucino PC runs Linux as host OS while can run any Operating System as guest OS. Users are offered a fully virtualized platform that allows them to select the Operating System that they prefer for their research. This architecture allows users to develop their protocols and applications at the User level as well as in the operating system kernel. The physical hardware accessories such GPS, WLANs, etc are also virtualized through the $C-VeT$ software platform and shared by all the running virtual machines. Experiments that may require changes in the MAC parameters are supported using MadWiFi [3] but they require an exclusive booking of the mobile nodes for the experiment duration.

III. Demo Setup

In our experiments we use 5 cars equipped as described above. In this set of experiments the multi-hop connectivity is achieved using the OLSR Daemon application [2]. We perform 2 sets of experiments: Peer to Peer File Sharing and Video Streaming. Additionally, we constantly monitor the network characteristics and performance using the $C-VeT$ hardware and software monitoring infrastructure. In the following we describe the main features of our experiments.

A. Network Monitoring

Each car is connected to a Monitoring Station through the backbone channel described in Section II-A. Periodically, each car collects the following statistics:

- **GPS Information**: using the GPS receiver the node gets its own time-stamped position, altitude and speed.
- **Routing Information**: the OLSR daemon constantly updates the routing table using the *Hello Messages* exchanged among the nodes. Therefore processing its own routing table the node discovers its one-hop neighbors.
- **Network Activity**: using the DU Meter application the node measures the upload and download throughputs, as well as the number of incoming and outgoing TCP connections.

This information is encapsulated in a single packet along with the sender's IP address and forwarded to the Monitoring Station. The Monitoring Station receives the packets coming from the cars and stores the information in a MySql database. Moreover, the Monitoring Station runs a web server that dynamically plots the position of the nodes in the network on Google Maps along with the network statistics.

B. Peer to Peer File Sharing

We investigate the behavior of the Peer to Peer (P2P) protocol BitTorrent [5] in VANETs. For this purpose we use a free BitTorrent client, Azureus [4]. This client is an open source java application and allows the use of a distributed tracker, which is an essential feature to distribute files amongst nodes without internet tracker support.
We perform two different tests, each one representing a possible vehicular network application:

- **Large Content Distribution**: in this scenario one of the cars has access (e.g. downloading it from an access point) to some content of interest, such as a movie trailer. This car may also act as a source for distributing the content which can be downloaded by the other peers in the network.
- **Small Content Distribution**: in this second scenario, the nodes will share a consistent number of small files, such as images or text files from peers in the network.

The two scenarios discussed above show the main issues related to P2P file sharing in VANET in diametrically different cases for file size and content type.

C. Video Streaming

We hypothesize a situation where a police car attempts to get some of the videos collected from the cars. For this purpose we develop two applications, one on the police car (Control) and the other on the rest of the cars (Trigger). The trigger application waits for a query from the Control application. When the query is received it starts a VLC [7] server that streams the video to the police car via a unicast UDP flow. This allows two possible applications:

- **Emergency Control**: In case of an emergency the police car can issue a broadcast request to all the cars that are at most 2 hops away. This provides the police car with a comprehensive view of the whole situation.
- **Suspicious Vehicle**: this presents a case where the police car suspects a vehicle for illegal activities and requests the video to better understand the situation.

References

[1] A. Capone, S. Napoli, and A. Pollastro. Mobimesh: An experimental platform for wireless mesh networks with mobility support. *Proc. of ACM QShine 2006 (WiMESHNets 2006, Workshop on "Wireless mesh: moving towards applications")*, 2006.
[2] T. Clausen and P. Jacquet. Optimized link state routing protocol (olsr). *RFC3626*, 2003.
[3] madwifi.org.
[4] www.azureus.com.
[5] www.bittorrent.com/protocol.html.
[6] www.vehicularlab.org.
[7] www.vlc.org.