C-VeT, the UCLA Vehicular Testbed: An Open Platform for Vehicular Networking and Urban Sensing

Paolo Lutterotti*‡, Giovanni Pau*, Daniel Jiang†, Mario Gerla†, Luca Delgrossi†
*Computer Science Department, UCLA, CA, USA
†Daimler Research Center, Palo Alto, CA, USA
‡Istituto Superiore Mario Boella, Torino, Italy

***Abstract*—In the last few years Wireless Vehicular networks have been in the limelight for their potential in many human applications such Road Safety, Info-Mobility, Infotainment, Environmental Monitoring, etc. The research community responded to the new emerging needs designing a number of applications, algorithms, and protocols to cope with the vehicular environment. Differently from tactical ad hoc networks, vehicular networks present several new challenges including: high-speed mobility of the nodes, (resulting in a relatively short contact time), constrained mobility models, harsh propagation environment, high radio interference, and frequent network disruption in urban sparse traffic. Hence, a totally new network paradigm is needed.**

In this position paper we argue Vehicular Testbeds as the ideal playground to study the behavior of a new class of vehicular applications and protocols in an highly realistic environment; yet maintaining control on the experimental scenarios and the ability to perform large scale experiments through the integration of testbeds and simulation tools.

I. Introduction

Vehicle communications are becoming increasingly popular, justified by navigation safety requirements and enabled by investments of car manufacturers and Public Transport Authorities. With the expansion of applications scenarios from safe navigation and collision avoidance to content distribution, emergency operations recovery (natural disasters, terrorist attacks, etc.) and urban pervasive sensing, a brand new set of services will emerge in the Vehicular Network. For example, efficient, congestion protected, priority oriented broadcast of vehicle alarms; vehicle to vehicle routing robust to disconnections, possibly geographic and capable of exploiting the Infrastructure; use of MIMO radios for diversity protection and capacity enhancement. The new vehicular applications will also place new service requests to the Internet Infrastructure. Namely: urban friendly addressing for example, geo addressing as opposed to (or in addition of) traditional IP addressing; directory service support, service discovery, mobile resource monitoring, and mobility management.

New standards are emerging (DSRC[1], IEEE 802.11p[2]) and small scale (often single vehicle) testbeds have demonstrated the feasibility of car to car communications to enhance safety, prevent intersection crashes and rapidly propagate accident reports. The availability of on board radios and dedicated spectrum will pave the way to a host of new applications for the *"vehicle grid"*. These emerging applications span many domains: from office on-wheels (e-mail, file transfers, work groups) to entertainment to passengers (mobile internet games, multimedia, news), e-commerce, crime investigation, civic defense, etc. Many of these applications are location *"aware"*, i.e., they relate to neighborhood resources and services (e.g., restaurants, movie theaters, etc). Others involve close *"cooperation"* among drivers, leading for instance to creation, *"temporary"* storage and *"epidemic"* distribution of sharable content. Examples of the latter class include the collection of *"sensor data"* by cars acting as *"mobile sensor platforms"*; the sharing and streaming of files using the *"Car-torrent"* P2P software [3] and; the creation/maintenance of massively distributed commercial, entertainment and culture information data bases.

Most of the above protocols applications were developed and tested via simulation - yet several features of the urban environment (mobility, propagation) cannot be properly simulated. For some properties (e.g., radio propagation, MIMO support, user reaction times, etc) the only realistic validation is through a vehicular testbed. Current testbeds, as later discussed, are too small and too sparse to create ad hoc vehicle networks of adequate size to properly validate the emerging P2P applications. In order to properly assess the new Vehicular Grid services, i.e. to measure their demand on the Internet Infrastructure and their use of current wireless media such as Cellular, WiFi, WiMAX, Mesh Networks, Satellites, etc.; we describe here the UCLA Campus Vehicular Testbed (C-VeT). The Testbed will have (at maturity) a reasonable vehicle density (up to 60 nodes). It will carry V2V vehicle communications as well as V2I communications, and at the same time it will interface to the Internet infrastructure, allowing shared access to remote experimenters via a web-based interface.

The C-VeT testbed will enable the experimental evaluation of protocols and models that are at the core of vehicular networking. In particular:

- Radio and MAC layer performance assessment (e.g., download/upload capacity at Infostations at various speeds; car to car achievable data transfers).
- Efficient use of multiple 802.11p channels (control and data; prioritization of channels and data, etc); coexistence

of critical and infotainment traffic.

- Network protocol testing: new network protocols will be deployed to support emerging applications (e.g., epidemic dissemination, scoped broadcast, redundant forwarding control, multihop routing, network coding, congestion control, etc); the testbed will allow evaluation of such applications in different traffic/motion/channel scenarios. It will also allow comparison of ”competing” schemes possibly, side by side using virtualization.
- Traffic and Mobility model experimental validation. The performance of most vehicle applications depends critically on vehicle density, motion pattern and correlation, making the testbed an ideal platform to exploit the characteristics of realistic urban mobility and traffic conditions. The C-VeT testbed is deployed *in the wild* to allow researches to study the ”natural” mobility of vehicles. The UCLA facility management vehicles, indeed, are not mobility constrained, they operate in response of the campus needs and requests (e.g. Campus Mail, building maintenance, etc). One of the goals of the testbed is to develop representative, synthetic models of motion, channel and traffic that can be validated through actual urban measurements.

The overarching goal of the C-VeT testbed is to lay the grounds for a shared platform to develop, and evaluate algorithms, protocols, and applications for vehicular networks. This goal is achieved using dozens of UCLA Facilities vehicles integrated by private vehicles and infrastructure nodes; deploying network and experiment monitoring software for efficient use of C-VeT from remote sites and by multiple simultaneous users; investigating emerging vehicular protocols and applications and paving the way to their successful commercialization, and; seeking the collaboration with synergistic projects Nationwide. Finally to support the need to test the protocols in large scale *“repeatable”* scenarios, we will *“augment”* C-VeT with popular emulation platforms including: the CMU wireless Emulator [4] (to emulate propagation) and; scalable hybrid-simulation platforms, such as the Qualnet simulator stemmed from the NSF funded WHYNET project [5], and the ORBIT emulation platform at Rutgers [6][7], [8], [9], [10], [11], [12]. In particular, C-VeT will be fully integrated with Qualnet providing a seamless hybrid simulation platform thus allowing users to develop a single code-base for both actual testbed and hybrid simulation experiments.

The reminder is organized as follows: in section II we discuss the current testbed initiatives available in the field and make the case for a new Vehicular Testbed. The C-VeT architecture and component details are introduced in section III, while in section IV we report some preliminary results obtained for p2p applications. Finally section V concludes our paper.

II. Related Work

In the last few years Wireless Vehicular networks have been in the limelight for their potential in many human applications such as Road Safety, Info-Mobility, Infotainment, Environmental Monitoring, etc. The research community responded to the new emerging needs designing a number of applications, algorithms, and protocols to cope with the vehicular environment. Differently from tactical ad hoc networks, vehicular networks present several new challenges including: high-speed mobility of the nodes, (resulting in a relatively small contact time), constrained mobility models, harsh propagation environment, high radio interference, and frequent network disruption in sparse traffic. Hence, a totally new network paradigm is needed. While recent projects such as the CartTel and Cabernet projects at MIT[13], [14], [15], [16], [17], the Dome and DieselNet projects at the University of Massachusetts Amherst [18], [19], and the Microsoft VanLan Project [20], [21], [22], made an effort to provide a platform to test vehicular protocols and algorithms; no open platform specifically designed for vehicular research has been launched. C-VeT in contrast will be OPEN to remote experimenters. CarTel consists in a distributed, mobile sensor platform able to support a broad range of sensing and communication applications. CarTel nodes are deployed with a partial leverage of 27 cars of a local limo company (PlanetTran). An embedded platform on the vehicle interfaces with a variety of sensors in the car, processes the collected data, and delivers it to an Internet server. CarTel uses wireless networks opportunistically. It uses a combination of WiFi, Bluetooth, and cellular connectivity using whatever mode is available and working well at any time, but shields applications from the underlying details and network disruptions. DieselNet consists of 40 buses each with a*“Diesel Brick”*, which is a computing platform based on a HaCom Open Brick computer. The node is connected to three radios: an 802.11b Access Point (AP) to provide DHCP access to passengers and passersby, a second USB-based 802.11b interface that constantly scans the surrounding area for DHCP offers and other buses, and a longer-range 900MHz radio for remote monitoring and management. Each bus has also a GPS device to record times and locations. A custom software developed by the UMAS Amherst allows to take mobility and connectivity traces, as well as to manage the AP-to-bus, and bus-to-bus connectivity. The VanLan project deployed by Ratul Mahajan at Microsoft Research consists of 11 WiFi basestations and 2 vans that circulate around the Microsoft Seattle Campus during day operating hours. The VanLan project targets mechanisms that enable inexpensive, high bandwidth connectivity to moving vehicles in urban areas through opportunistic hopping between open WiFi base stations.

Finally, the ORBIT platform developed at Rutgers University [6] provides a large scale emulation platform for wireless ad hoc networks. ORBIT features 400 fixed nodes installed in a large facility at Rutgers. The ORBIT nodes provide a virtualized facility to perform experiments in a controlled environment that feature simulated mobility and simulated interference through a wireless jammer [7], [8], [9], [10], [11], [12]. The ORBIT testbed represents a useful facility but unfortunately does not provide actual real-world mobility nor actual propagation features in urban scenarios.

On the VANET commercial scenario we mention Dash Express, a P2P network that supports the Dash Navigation System and became operational in early 2008. Similar to CarTel, subscribers (i.e., Dash Express nodes) monitor their travel delays and report them to an Internet portal. The cellular network is used to connect to the portal (open WiFi access points will be used when available). The gathered traffic information is used for traffic flow analysis and routing recommendations. Dash Express users pull real-time traffic information via GSM or WiFi. Dash Express is NOT intended to be a testbed. However, it is reported here as example of one of the first known commercial VANET applications.

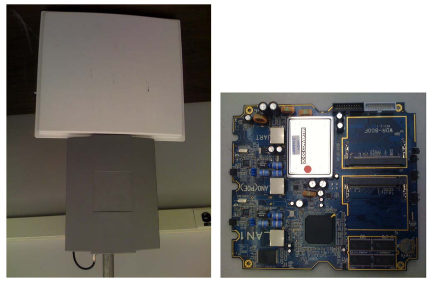
While the three early testbeds offered an important initial opportunity to perform a number of key studies, nevertheless they are not designed to provide the type of open platform that is required for collaborative research in V2V communications and Urban Sensing. In particular, the above platforms lack the virtualization mechanisms that enable sharing of the infrastructure by several researchers independently (at the same time insulating each user like in a sandbox). In addition, a set of standardized mechanisms for testbed control (i.e. user software deployment, developing platform, development emulation platform, etc.) are not included in the initial deployments of the above testbeds. This is understandable since they were designed to target a specific class of users and to be operational in a short time. Finally, the vehicles used in the initial testbeds led naturally to intermittent connectivity scenarios, thus enabled only delay tolerant applications. In DieselNet and in VanLan the use of Vans and Busses respectively leads to very stable, largely predictable mobility patterns and to a delay tolerant approach. In CarTel the use of city cabs better matches the mobility of an urban scenario but the 27 vehicles involved in the testbed are too few to maintain a connected network over the area covered thus enabling only applications that are highly delay tolerant. Our proposed testbed is the first vehicular testbed that will offer reasonable connectivity (with up to 30 facility vehicles over approximately a 1 square mile area) and will be OPEN to outside experimenters. Both of these features are critical for the protocols and applications we are interested in evaluating and deploying on the testbed.

III. The Campus Vehicular Testbed - CVeT

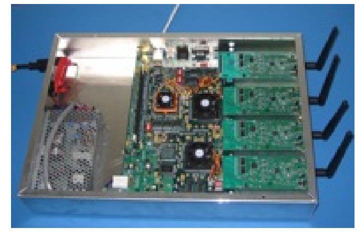
Our goal is to provide an open platform to support Vehicular Networks, Urban Sensing research and related applications. We inspired ourselves to the pioneer work done by Larry Peterson and Tom Anderson with Planet Lab[23]. In particular we want to build an always on, fully virtualized, web-accessible, sensor equipped testbed infrastructure. In our view the UCLA campus, with its 10 acres of urban development, features the scenarios, propagation, and communication challenges typical of a city, in a realistic manner but yet relatively small-scale. In particular the *C-VeT* will provide:

- A fully virtualized platform that runs both Linux based and Windows based operating system with full insulation between the sandbox and the ability for the user to re-design new protocols and algorithms such as, for instance, Routing protocols. C-VeT will also feature fully programmable fpga-based MIMO radios thus enabling MAC layer redesign. This feature will be key for the MAC researchers.
- A Campus Wide Mesh network developed using OPEN WRT and optimized for the integration in the Vehicular network, will help cope with the network disruption and enable, opportunistic, interactive and delay tolerant experiments.
- 30 Facility Management vehicles equipped with the C-VeT engine, will provide an always-on platform to run experiments, collect traces and measurements. The facility Management vehicles perform both routine and on-demand trips, resulting in several mobility patterns thus emulating the traffic behavior in a real city.
- 30 Campus Van Pools, equipped with the C-VeT-Census platform that will survey the environment gathering Traffic, Air Quality, and stereoscopic images. The aim is to build a large database that could enable new micro-pollution models and provide an image database to be used in vision based environment survey.
- A number of downloadable pre-configured virtual appliances to allow users to develop their protocol in house with the same software configuration of the testbed nodes.
- A large scale emulator that will allow users to debug their algorithms and protocols on the same environment of the actual C-VeT nodes but with emulated MAC and Physical layers components developed using Qualnet.
- A robust web interface that will manage the users and deploy the experiments in a streamlined fashion. The Web server will provide the front-end for a number of services and tools designed to help the users focus on the research issues instead of testbed. In particular, we will develop a set of services to deploy the experiments and a support to gather the data. We will provide a number of API to the low level interfaces and virtualize the hardware components. For instance we plan to extend the MadWiFi support to the Virtual Machines creating a virtual MadWiFi layer[24].
- The ability to develop algorithms, applications and protocols that directly *operate at Layer 2* using RAW sockets and TUN/TAP mechanisms for both Windows and Linux OS. Recent Research [25] shown that the TCP/IP suite may not be the most appropriate choice for vehicular networks and a *ground-up protocol stack redesign* is needed.
- An organized live database of mobility traces, sensed environmental data, road traffic information, Vehicle Can-Bus statistics, MAC layer statistics (through MAD WiFi) and Physical Layer statistics taken using the MIMO APIs when appropriate. This data collection will be offered to the research community and complement other active trace collection such as the CRAWDAD database[26].

(a) C-VeT Mobile Node, WiFi interface and GPS Sensor



(b) C-Vet Mesh Node



(c) Silvus Network SMC200 Programmable MIMO platform

Fig. 1. C-VeT Hardware Components

We designed the testbed using a top-down approach and the whole system can be described through a number of relatively simple building blocks: *the C-VeT mobile node*, *the C-VeT mesh node*, *the C-VeT-Census platform*, *the Web based Control Center*, and *the Emulation platform*

The C-VeT infrastructure is designed to provide an always-on facility for research in wireless vehicular network. To achieve this goal we chose to install our equipment in the UCLA Campus Facility management and Van Pool vehicles, this cars and vans are driven everyday to fulfill the campus needs and perform both routine and non routine tasks. In addition, to the 24/7 facility, will be always possible to add on-demand vehicles *equipping on the fly* students and staff cars with the C-VeT mobile nodes. We argue this is a perfectly suitable strategy to handle any special need that may arise during the C-VeT life.

A. The C-VeT mobile node

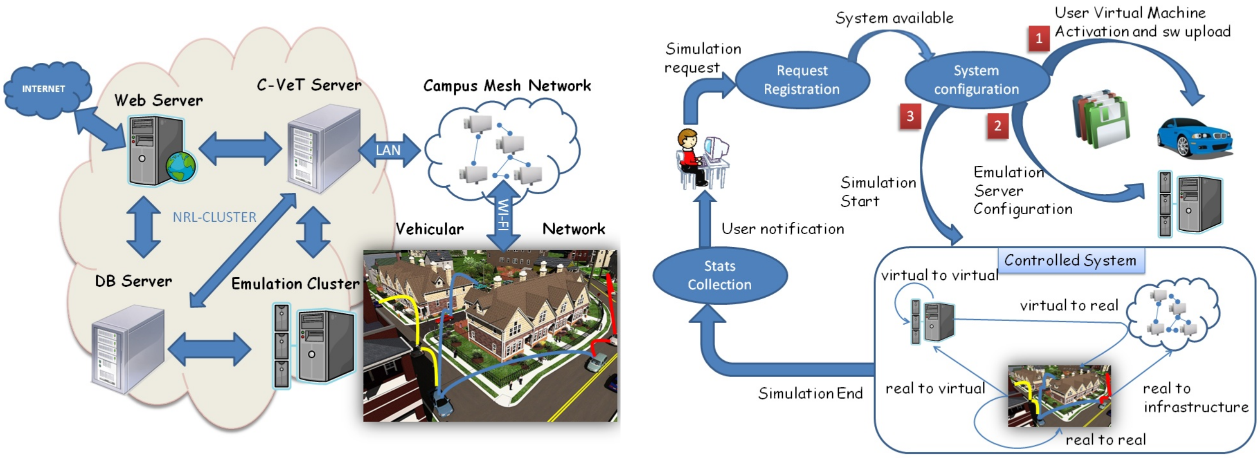
Each vehicle will be equipped with a C-VeT mobile node (Figure 1a). The node computing engine is based on industrial strength Cappucino PC powered by an Intel Dual Core Duo processor at 2.5GHz, 2GB of RAM, 320GB Disk. The Hard drive as well as the other internal parts are rugged to sustain the physical stress of being mounted on the top of vehicles (i.e. large temperature jumps, street bumps, etc). The computing engine is connected to the outside world using 3 Wireless Interfaces: one IEEE802.11a/b/g/n based on the Atheros AR9160 chipset, one IEEE802.11p interface based on a Daimler-Benz customized chipset and additionally a standard Bluetooth interface is made available. Few vehicles will be equipped with programmable Silvus SC2000 MIMO platforms (4x4 configuration) that will provide full access to the physical layer and enable a new generation of experimental MAC layer research (Figure 1c). Silvus Networks is committed to provide an open physical layer APIs thus enabling innovative experimental research at the MAC layer. Finally two vehicles will be also equipped with 3G interfaces to enable opportunistic routing to the Internet thorough a mobile node.

The C-VeT nodes are instrumented with a customized sensor platform designed to provide a flexible data collection. In particular, each C-VeT node will be instrumented with an Infra-Red based CO2 sensors and with electrochemical CO sensors. Additionally the vehicles will have a SIRF III or Ublox based GPS sensor, a mega-pixel camera, temperature, and humidity sensors. Few cars will be equipped with OBD-II readers that enable the access to the in-car instrumentation, and few other vehicles will be equipped with a CANBUS interface (with the support of Daimler-Benz): this will allow to perform close-loop control on some of the vehicle actuators. Using the C-VeT cars as mobile air quality sensors will enable a new wave of atmospheric research aimed at the use of mobile sensing agents to study the air quality at the neighborhood level as shown in [27], [28], [29], [30] for this reason we will equip part of the fleet with high performance **exhaust particulate** sensors DC2000CE by echocem [31] thus being the first testbed able to support the currently leading research in micro-climate air quality.

Our general vision is to create an infrastructure that can be shared by several research groups at the same time yet allowing exclusive use on demand. To achieve this goal we emphasized the virtualization of the CeVeT mobile nodes and related sensing platform. The node *software architecture* features the resource virtualization at its core. Each C-VeT node features a layered software architecture that allows seamless resource sharing among the Virtual Machines. In particular, each node is equipped with a Gentoo Linux distribution with kernel 2.6.25 patched with the XEN virtualization hyper-visor installed directly on the naked hardware; we call this *Host Machine*. This layer provides the virtualization engine and the services to the virtual machine. At this layer we will develop a number of virtualized hardware drivers able to provide the low level information to the Guest os in the virtual machines. In particular, we will virtualize: *(a) the GPS Sensor*, *(b) the external sensing platform*, *(c) the MadWiFi interface*, *(d) the in-vehicle sensors (i.e. CANBUS)*.

The *sensor platform virtualization* will be performed exposing the sensor data from the actual hardware to the virtual machines using a virtual network interface. Similarly we will handle the actuators managing the system coherence when needed.

The *MadWiFi virtualization* will be performed at the kernel level and lead by Christian Benvenuti, author of the O'Really book Understanding Linux Network Internals [32]. The approach consists in building a virtual MadWiFi module



(a) C-Vet System Architecture

(b) Usage Case

Fig. 2. C-VeT Software Architecture

to be installed in the Guest Machines. The virtual module will provide real-time access to the NIC statistics, a coherent view of the hardware status, manage the lock and release of the hardware resources when users perform a write on the MadWiFi parameters (i.e. keeping the NIC interface status coherent when switching between two subsequent users).

Each C-VeT user will get full control of its Virtual Machine and will be able to perform any operation at the virtual machine level: this includes changes on the Guest operating system kernel. C-VeT will support both Windows and Linux based virtual machines and for both platform we will provide a number of services designed to help researchers to keep focus on the research challenges rather than the testbed complexity. In particular we will expose TUN/TAP interfaces that will allow to send packets at Layer two without the effort of writing a kernel module. In linux this will be performed using the standard Linux TUN/TAP support while in Windows this goal will be achieved thanks to a Virtual driver interface made available by Microsoft Research Cambridge (VRR -Driver) [33], [34]. The XEN hyper-visor will guarantee the fair-share of the computational resources among the virtual machines and we will enable throttle mechanism in the host machine kernel to reserve a fair amount of bandwidth for each virtual machine at the various network interfaces.

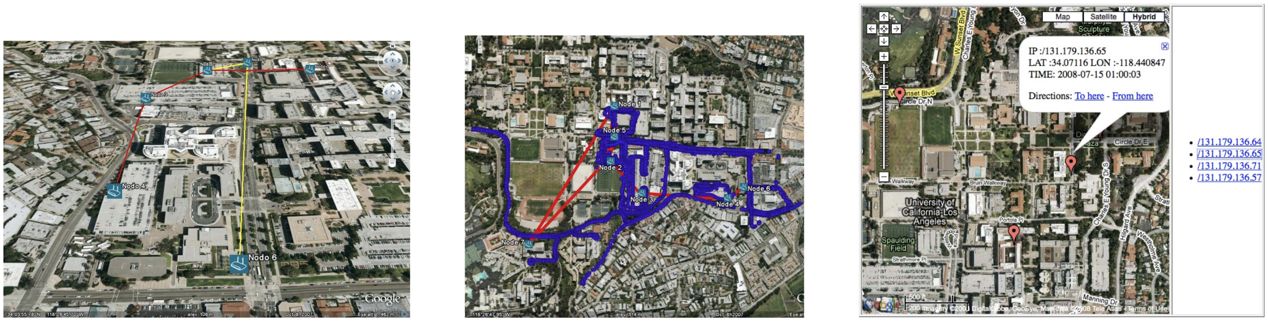
A middleware layer will be implemented in the Host Machines (natively on the host OS) to enable testbed wide services and the gathering information for long-living experiments (i.e. mobility traces, and air quality data). In particular, this will include a number of APIs designed to perform *(a) Authorization and authentication services*, *(b) Seamless experiment deployment*, *(c) Node management*, *(d) Data gathering* (i.e. mobility traces, 1-hop neighbor, etc), and *(e) Testbed maintenance* (i.e. reboot a node, upload new software releases, suspend a Virtual Machine, etc).

B. The C-VeT mesh node

The C-VeT mesh node is based on the MobiMesh [35] hardware (see Figure 1b). The C-VeT mesh nodes feature Open WRT OS and the hardware supports the MadWiFi (Atheros Chipset) thus creating a fully integrated network with the mobile nodes. The fixed infrastructure will be installed on the roof top of UCLA buildings, aiming at campus coverage and the integration with the current campus wireless infrastructure. The resulting mesh network will act as a programmable platform to allow opportunistic Internet access from the vehicles, as well as a key point of the C-VeT infrastructure, enabling cross-campus communications between cars and a control channel to the vehicles. The Mesh network configuration will be performed via web and particular routes may be configured on the demand by the network administrator to perform particular set of experiments. The C-VeT integrated approach that includes the infrastructure component, as well as the vehicular component, broadens the experimental scenarios and allows to set up a control channel for the testbed management and monitoring. In the initial phase, we will cover the south campus. This will lead to an initial base of 6 mesh points and 8 vehicles. The initial campus coverage map is shown in Figure 3b, while the initial campus mesh backbone is depicted in Figure 3a.

In particular, the mesh network will be configured to offer a ”blind-pipe” for cross campus connectivity and Internet forwarding. For instance a vehicle on the south part of the campus could be enabled to use the mesh network to connect to another vehicle on the north side of the campus, or could be enabled to get internet access. On the other side of the coin, a user that wants to connect to the vehicular testbed to run an experiment, implicitly uses the mesh network to perform the experiment deployment and monitoring; ventually he or she can request the mesh network to be part of the experiment, or can just use the V2V testbed.

In order to achieve seamless integration between the CVeT-



(a) Initial C-VeT Mesh Backbone (b) Initial C-VeT Mesh Coverage (c) Preliminary Web based Control Center

Fig. 3. C-VeT Network Deployment

Mesh and the Vehicular network components we will develop Layer 3 and Layer 2 routing and VLAN mechanisms. The first set of mechanisms will allow *network layer* routing between the moving vehicle and the fixed nodes thus enabling a communication cross campus, as well as cross the Internet. The Layer 2 VLAN mechanisms instead are designed to put a subset of the vehicles (selected by the experimenter) in the *same broadcast domain* at L2, ignoring the fact that several infrastructure nodes may be in the actual physical path: this approach is useful for experiments that require an higher number of nodes. Within the scope of C-VeT we will perform a modification of the Open WRT kernel to include the new routing and L2 features. The software will be developed in house in collaboration with the hardware provider MobiMesh and a leading linux kernel developer [32].

C. The C-VeT-Census platform

The *C-VeT-Census platform* is a tailor maid sensor platform designed to be connected to the C-VeT mobile nodes through USB or RS232 as well as to be connected to a lightweight PDA for mere data collection. In particular the sensor platform will feature: GPS, Camera, temperature sensors, humidity sensors, accelerometers, CO2 Infrared Sensors, CO sensors, high definition particle sensors [31]. The majority of the sensors, with the exception of the digital cameras, produce an analog singnal either in current or voltage that is then sampled using a 16bit ADC and further converted to a RS232 signal. The *C-VeT-Census* platform is the primary source of data and will be used to feed the database with a set of geo-referenced and time-rerenced data, In our intention this will provide a growing database of information that will be useful for a number of researchers in the ITC area but also researchers from environmental and atmospheric sciences as well as from transportation engineering. In particular we will develop a mobility traces database, a pollution database, and a image database for 3D localization and reconstruction.

D. The Development and Emulation platform

The *C-VeT Development and Emulation* platform aims at support the research during its early development stage and to increase the scale of the testbed introducing simulated nodes in a hybrid simulation fashion[36], [37], [38]. In particular, we will provide to our user a development platform based on the same exact hardware installed on the vehicles with the aim of allowing a streamlined development process. We will install an initial lot of 10 virtual instances of C-VeT Mobile nodes in our laboratory to serve as emulator and development facility open to the users. The mobility as well as the positioning will be re-playing the traces from the testbed and synthetic, while the communication channel will be simulated using Qualnet.

The Qualnet simulator will be also installed in a cluster of 5 servers based on the Intel Xeon Processor X7460 (6 core, with 16MB of L2 Cache). Each server will be equipped with 64GB of RAM and they will share a 5 Terabyte of disk through a gigabit LAN. The disk facility will be implemented in RAID 5 technology with an high end-controller that provides cache on write. This system will provide the computational power to perform real-time hybrid simulation with several hundreds of nodes. Our preliminary results show the current architecture with 2 Quad Core Xeon can simulate up to 2000 nodes in realtime, we expect the new computing platform to scale up to 10,000 nodes in realtime.

The C-VeT emulation platform is designed to support the research cycle and allow to perform large scale experiments in a controlled environment (see Figure. 2b). In particular, we designed the hybrid-emulation platform to be transparent to the users. The virtual appliances deployed Cappucino PC located at the Network Research Laboratory at UCLA, will be instrumented with a kernel level virtual network interface that will actually route the packets to the Qualnet simulator instead of the actual network; at the same time packets can travel through the actual campus testbed. The model we envision is that the end-user applications will run on the actual C-VeT mobile nodes, and the simulated network will perform the In-network relay. The users will develop their application ignoring the presence of the simulations and take advantage of the simulation facility when designing large scale scenario.

E. The Web Based Control Center

The *C-VeT Control Center* performs vital tasks to ensure the correct integration of the various C-VeT components. In particular the control center offers a WEB-Based interface over the Internet where users can *register*, *access to the development tools*, *deploy and schedule experiments*, *access* *the C-VeT databases*, *monitor the system status in realtime*, and *perform system maintenance*. In particular, the control center will have a front end based on the apache web server [39] that will allow users to enjoy a friendly interface while the service back-end will be built integrating several web-tecnologies based on java containers [40], and database engines such as MySQL or PostgreSQL (see Figure 3c for an initial example of the C-Vet Monitoring interface). The control center will play a key role in the experiment life-cycle as shown in Figure 2b. We will develop the control center front-end and back-end functionalities, including the experiment deployment and data collection mechanisms, which will be designed to be disruption tolerant and transactional. The aim is to develop a software layer able to cope with the frequent network disruption and the possible system disruptions (i.e. the car is turned off and the battery is running low so a shutdown is forced). We believe that being able to guarantee coherence in the transactions within the testbed is essential for producing high quality results.

IV. Preliminary Experiments

We performed an initial set of experiments to assess feasibility of multi-hop communications in a vehicular environment. In particular we designed and deployed two experiments: (i) a video streaming experiment; (ii) a P2P content dissemination experiment. In both experiments we studied the feasibility of multi-hop vehicular communications in mobile urban scenarios.

A. P2P video streaming on the go

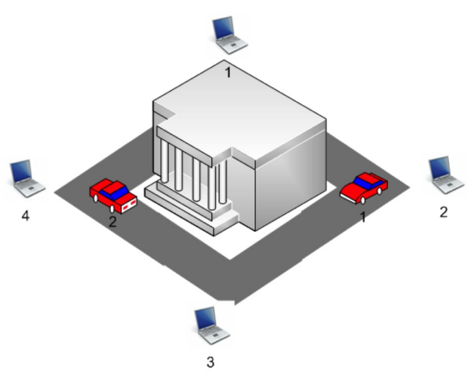


Fig. 4. P2P Video Streaming Experiment with 2 Vehicles

In this experiment we set up a simple and very controlled scenario to understand if the current networking protocol stack supports the communication under vehicular mobility[41].
In particular, we setup a vehicular network around the UCLA computer science and Engineering department between Boelter Hall and the Engineering IV buildings. We placed 4 fixed nodes at the 4 corners of the building and two mobile vehicles moving in clockwise around the building. The vehicle speed has been studied to make sure that under no circumstances the cars had a direct link. The nodes in the network all run a Windows XP SP2 with the INRIA OLSR implementation [42][43] configured for vehicular environments with an hello interval of 100ms. One of the vehicles had a web-cam installed and the movie captured was transmitted over the network using the VLC [44] streaming server. Node 3 (see figure 4), using the VLC client, was requesting the video stream thus simulating a live TV show transmitted from the Car. The VLC client playout buffer size was set in 1MB in order to reduce video interruptions and being able to study the loss in this configuration. The network topology, depicted in figure 4, is designed to force the routing table updates every half circle around the building in order to stress the network stack. Experimental results show that about 10% of the video frames were lost according to the data reported by the VLC video client and TCPDUMP [45] (see figure 5). The video was encoded as MPEG2 video stream at Variable Bit Rate. Visually the video appears to be smooth for most of the experiment though two remarkable interruptions have been registered at the receiver side. During the experiment all

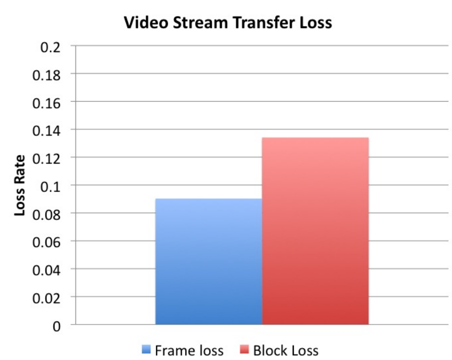


Fig. 5. P2P Video Streaming initial results - Frame Loss

the traffic was logged on disk using TCPDUMP. The off line video analysis shows that the received video is shorter than the original due to the frame loss (feel free to see the video comparison at: **http://www.vehicularlab.org/demovideo.do?idSection=87** last video on the page). A further investigation showed, as expected, a perfect correlation between the routing changes and the loss events at the packet level, thus pointing at new challenges for the routing in vehicular networks.

B. P2P Content Dissemination

In the beginning of Fall 2008 we performed, at UCLA, a second set of experiments aimed at exploring the data dissemination in vehicular networks. In order to achieve an higher density of vehicles we employed 8 cars. The drivers were instructed to follow a particular path in the roads around the campus as shown in figure 6a. The complete picture of all the driving path has been designed to emulate urban traffic pattern in a 2 blocks area. The vehicle, indeed, had a relatively small inter-contact time and once again the routing was challenged. In this experiments we used DELL D530 laptopos equipped with GPS sensors based on the SIRF3 Chipset. In Addition, each car was equipped with an off-the-shelf IEEE802.11g network interface connected to an 8Dbi

(a) Mobility Patterns

(b) Preliminary Results: Download Time (per vehicle)

Fig. 6. P2P Dissemination Experiment with 8 Vehicles

gain external antenna. We used a Windows XP SP2 operating system and the INRIA OLSR implementation properly configured for vehicular networks. We used a modified version of Azureus[46], in order to disseminate a large 130MB file placed on the Internet throughout the vehicular network. The source file was sitting in an Internet-accessible server at UCLA, the vehicles were connected to the Internet through a fixed access point located at the intersection between Le Conte and Gayley avenues (see figure 6a). Vehicles 1 and 2 were never in the AP one-hop neighbor and vehicles 3 and 5 were in the AP range only once every two circles.

Figure 6b shows the download time for each of the eight cars. In particular, the average download time for the 130MB file is 63 minutes, with a standard deviation of 20.042; the fastest download was performed by Car #6 in 30 minutes and 15 seconds. Vehicles #6 and #7 are the fastest to complete the download. This is mainly due to the path assigned that leads to more chances to get data from the source directly and to a higher probability to encounter other vehicles (their path is the shortest see figure 6a; ”Le Conte”→”Gayley”→”Weyburn”→”Broxton”, clockwise and counterclockwise). The above path, indeed, is both the shortest circle hence resulting in a more frequent passage under the AP. Additionally the above path shares three out of four roads with 3 other paths resulting in an higher probability to encounter another vehicle.

V. Final Remarks

We believe vehicular networks will assume an increasing key role in several areas of the human activities such as Intelligent Transportation, Environmental Protection, Mobile Entertainment, Safe Navigation, and National Security, just to name few. Is also clear that car manufacturers will lead the road safety and transportation innovations while they will leave informational applications, intelligent navigation, and entertainment applications to OEMs and aftermarket manufacturer i.e. Garmin, TomTom, etc.

However, in order to provide reliable, efficient, effective services and applications, a new communication paradigm is needed. The current protocol stack, indeed, is unfit to cope with the high speed mobility, harsh propagation scenarios, and disruption-prone environments typical of vehicular networks. In particular:

- The *Transport Layer* needs to be redesigned to gain efficiency in networks with higher loss rate that would harm the current TCP protocol. Brand new protocols are needed.
- New *Vehicular Routing Protocols* need to cope with high speed mobility and potentially large flat networks while achieving the integration with the current and future Internet.
- The protocol stack itself needs a to be redesigned with a clean-slate approach that leads to a shorter stack closing the gaps between the layers. Recent studies[25] shown that interactions between the layers lead to substantial performance degradations to be addressed when developing the next generation of vehicular protocols.

In this paper we described the architecture and features of the UCLA Campus Vehicular Testbed and argued for the key role of testbeds in studying vehicular networks. In our opinion, traditional approaches in developing and testing algorithms and protocols are not sufficient to describe the complexity and the details needed to design efficient and effective vehicular systems. Analytical models and simulations alone are no longer adequate tools to assist the design and development of vehicular protocols, as they do not consider the full range of variables present in a real scenario. In particular key factors such mobility and urban propagation are very hard to model and quite difficult to collect and analyze. Testbeds in contrast provide a realistic battlefield for new protocols and applications, allowing researchers to challenge themselves with the difficulties of a dynamic and harsh networking environment at the price of a smaller experiment scale.

Thanks

This work has been partially supported by the NATIONAL SCIENCE FOUNDATION through the grants SWARMS and OVERPROBE; bye the Italian Ministry for Research and Education through the GRANT DAMASCO; by Daimler North America, by UC DISCOVERY GRANT on P2P Video Streaming private sponsor STMicroelectronics.

References

[1] *DSRC Standards: What's New?*, ITS Standards Advisory, http://www.standards.its.dot.gov/Documents/advisories/dsrc_advisory.htm.

[2] D. Jiang and L. Delgrossi, "Ieee 802.11p: Towards an international standard for wireless access in vehicular environments," in *Proceedings of IEEE Vehicular Technology Conference 2008 (VTC 2008)*, Marina Bay, Singapore, May 2008, pp. 2036–2040.

[3] A. Nandan, S. Das, G. Pau, M. Gerla, and M. Sanadidi, "Cooperative Downloading in Vehicular Ad Hoc Wireless Networks," in *Proc. of IEEE/IFIP International Conference on Wireless On demand Network Systems and Services, St. Moritz, Switzerland*, 2005, pp. 32–41.

[4] G. Judd and P. Steenkiste, "Using emulation to understand and improve wireless networks and applications," in *Proceedings of 2nd Symposium on Networked Systems Design and Implementation (NSDI 2005)*, Boston, MA, USA, ay 2005.

[5] "http://chenyen.cs.ucla.edu/projects/whynet/," University of California, Los Angeles, http://chenyen.cs.ucla.edu/projects/whynet/.

[6] D. Raychaudhuri, I. Seskar, M. Ott, S. Ganu, K. Ramachandran, H. Kremo, R. Siracusa, H. Liu, and M. Singh, "Overview of the orbit radio grid testbed for evaluation of next-generation wireless network protocols," in *Proceeding of Wireless Communications and Networking Conference (WCNC 05)*, New Orleans, USA, Mar 2005.

[7] ——, "Overview of the ORBIT radio grid testbed for evaluation of next-generation wireless network protocols," *Wireless Communications and Networking Conference, 2005 IEEE*, vol. 3, 2005.

[8] M. Ott, I. Seskar, R. Siraccusa, and M. Singh, "ORBIT testbed software architecture: supporting experiments as a service," *Testbeds and Research Infrastructures for the Development of Networks and Communities, 2005. Tridentcom 2005. First International Conference on*, pp. 136–145, 2005.

[9] S. Ganu, H. Kremo, R. Howard, and I. Seskar, "Addressing Repeatability in Wireless Experiments using ORBIT Testbed," *Proceedings of TRIDENTCOM*, p. 153, 2005.

[10] M. Singh, M. Ott, I. Seskar, and P. Kamat, "ORBIT Measurements framework and library (OML): motivations, implementation and features," *Testbeds and Research Infrastructures for the Development of Networks and Communities, 2005. Tridentcom 2005. First International Conference on*, pp. 146–152, 2005.

[11] D. Raychaudhuri, M. Ott, and I. Secker, "ORBIT radio grid testbed for evaluation of next-generation wireless network protocols," *Testbeds and Research Infrastructures for the Development of Networks and Communities, 2005. Tridentcom 2005. First International Conference on*, pp. 308–309, 2005.

[12] K. Ramachandran, H. Kremo, M. Gruteser, P. Spasojevic, and I. Seskar, "Scalability Analysis of Rate Adaptation Techniques in Congested IEEE 802.11 Networks: An ORBIT Testbed Comparative Study," *World of Wireless, Mobile and Multimedia Networks, 2007. WoWMoM 2007. IEEE International Symposium on a*, pp. 1–12, 2007.

[13] "http://cartel.csail.mit.edu/doku.php#papers." [Online]. Available: http://cartel.csail.mit.edu/doku.php#papers

[14] B. Hull, V. Bychkovsky, Y. Zhang, K. Chen, M. Goraczko, A. Miu, E. Shih, H. Balakrishnan, and S. Madden, "CarTel: a distributed mobile sensor computing system," *Proceedings of the 4th international conference on Embedded networked sensor systems*, pp. 125–138, 2006.

[15] V. Bychkovsky, B. Hull, A. Miu, H. Balakrishnan, and S. Madden, "A measurement study of vehicular internet access using in situ Wi-Fi networks," *Proceedings of the 12th annual international conference on Mobile computing and networking*, pp. 50–61, 2006.

[16] J. Eriksson, H. Balakrishnan, and S. Madden, "Cabernet: A Content Delivery Network for Moving Vehicles," 2008.

[17] K. Chen, "CafNet: A Carry-and-Forward Delay-Tolerant Network," Ph.D. dissertation, Massachusetts Institute of Technology, 2007.

[18] A. Balasubramanian, R. Mahajan, A. Venkataramani, B. N. Levine, and J. Zahorjan, "Interactive WiFi Connectivity for Moving Vehicles," in *Proc. ACM SIGCOMM*, August 2008. [Online]. Available: http://prisms.cs.umass.edu/brian/pubs/balasubramanian.sigcomm.2008.pdf

[19] "http://prisms.cs.umass.edu/dome/index.php?page=umassdieselnet." [Online]. Available: \url{http://prisms.cs.umass.edu/dome/index.php?page=umassdieselnet}

[20] "http://research.microsoft.com/netres/projects/vanlan/." [Online]. Available: http://research.microsoft.com/netres/projects/vanlan/

[21] R. Mahajan, J. Zahorjan, and B. Zill, "Understanding wifi-based connectivity from moving vehicles," *Proceedings of the 7th ACM SIGCOMM conference on Internet measurement*, pp. 321–326, 2007.

[22] N. Banerjee, M. D. Corner, D. Towsley, and B. N. Levine, "Relays, Base Stations, and Meshes: Enhancing Mobile Networks with Infrastructure," in *Proceedings of ACM MobiCom*, San Francisco, CA, USA, September 2008. [Online]. Available: http://prisms.cs.umass.edu/mcorner/papers/RMB_mobicom.pdf

[23] "http://www.planet-lab.org/." [Online]. Available: \url{http://prisms.cs.umass.edu/dome/index.php?page=umassdieselnet}

[24] "http://madwifi.org/." [Online]. Available: \url{http://madwifi.org/}

[25] D. Hadaller, S. Keshav, T. Brecht, and S. Agarwal, "Vehicular Opportunistic Communication Under the Microscope."

[26] "http://crawdad.cs.dartmouth.edu/." [Online]. Available: \url{http://prisms.cs.umass.edu/dome/index.php?page=umassdieselnet}

[27] M. McCarthy, D. Eisinger, H. Hafner, L. Chinkin, P. Roberts, K. Black, N. Clark, P. McMurry, and A. Winer, "Particulate Matter: A Strategic Vision for Transportation-Related Research," *Environmental Science & Technology*, vol. 40, pp. 5593–5599, 2006.

[28] Q. Meng, B. Turpin, L. Korn, C. Weisel, M. Morandi, S. Colome, J. Zhang, T. Stock, D. Spektor, A. Winer *et al.*, "Influence of ambient (outdoor) sources on residential indoor and personal PM2. 5 concentrations: Analyses of RIOPA data," *Journal of Exposure Science and Environmental Epidemiology*, vol. 15, no. 1, p. 17, 2004.

[29] R. Lu, J. Wu, R. Turco, A. Winer, R. Atkinson, J. Arey, S. Paulson, F. Lurmann, A. Miguel, and A. Eiguren-Fernandez, "Naphthalene distributions and human exposure in Southern California," *Atmospheric Environment*, vol. 39, no. 3, pp. 489–507, 2005.

[30] W. Liu, J. Zhang, L. Korn, L. Zhang, C. Weisel, B. Turpin, M. Morandi, T. Stock, and S. Colome, "Predicting personal exposure to airborne carbonyls using residential measurements and time/activity data," *Atmospheric Environment*, vol. 41, no. 25, pp. 5280–5288, 2007.

[31] "http://www.ecochem.biz/Particles/DC2000CE.htm." [Online]. Available: \url{http://www.ecochem.biz/Particles/DC2000CE.htm}

[32] C. Benvenuti, *Understanding Linux Network Internals*. O'Really, 2005.

[33] M. Castro, G. O'Shea, and A. Rowstron, "Zero servers with zero broadcasts," *Proceedings of the 1st international workshop on Decentralized resource sharing in mobile computing and networking*, pp. 36–41, 2006.

[34] P. Ynard, "VRR: a study of design and implementation," Master's thesis, University of California Los Angeles, 2007.

[35] "http://www.mobimesh.it/index.php?lang=en." [Online]. Available: \url{http://www.mobimesh.it/index.php?lang=en}

[36] J. Zhou, Z. Ji, M. Varshney, Z. Xu, Y. Yang, M. Marina, and R. Bagrodia, "WHYNET: a hybrid testbed for large-scale, heterogeneous and adaptive wireless networks," *Proceedings of the 1st international workshop on Wireless network testbeds, experimental evaluation & characterization*, pp. 111–112, 2006.

[37] K. Xu, M. Takai, J. Martin, and R. Bagrodia, "Looking ahead of real time in Hybrid component networks," *Proceedings of the fifteenth workshop on Parallel and distributed simulation*, pp. 14–21, 2001.

[38] M. Kazantzidis, L. Wang, and M. Gerla, "On fairness and efficiency of adaptive audio application layers formultihop wireless networks," *Mobile Multimedia Communications, 1999.(MoMuC'99) 1999 IEEE International Workshop on*, pp. 357–362, 1999.

[39] "http://www.apache.org/." [Online]. Available: http://www.apache.org/

[40] "https://glassfish.dev.java.net/." [Online]. Available: https://glassfish.dev.java.net/

[41] E. Giordano, A. Ghosh, G. Pau, and M. Gerla, "xperimental evaluation of peer to peer applications in vehicular ad-hoc networks," in *he First Annual International Symposium on Vehicular Computing Systems*, Trinity College Dublin, Ireland, 2008.

[42] T. Clausen and P. Jacquet, "RFC 3626: Optimized link state routing protocol (olsr)," Oct. 2003.

[43] F. INRIA, *OLSR implementation http://hipercom.inria.fr/olsr/*.

[44] *VLC media player*, VideoLAN team, http://www.videolan.org/vlc/.

[45] X. Zhang, J. Kurose, B. N. Levine, D. Towsley, and H. Zhang, "Study of a Bus-Based Disruption Tolerant Network: Mobility Modeling and Impact on Routing," in *Proc. ACM Intl. Conf. on Mobile Computing and Networking (Mobicom)*, September 2007, pp. 195–206. [Online]. Available: http://prisms.cs.umass.edu/brian/pubs/zhang.mobicom.2007.pdf

[46] *Azureus - now called Vuze - Bittorrent Client*, Vuze, Inc., http://azureus.sourceforge.net/.