WWW@VDTN – A Web Browsing Application for Vehicular Delay-Tolerant Networks

João A. Dias[1], João N. Isento[1], Mário P. Pereira[1], Bruno M. Silva[1], Vasco N. G. J. Soares[1,2], Joel J. P. C. Rodrigues[1], Paulo Pereira[3], Augusto Casaca[3], Cristina Cervelló-Pastor[4], Javier Gallego[4]

[1]Instituto de Telecomunicações, University of Beira Interior, Covilhã, Portugal
[2]Superior School of Technology, Polytechnic Institute of Castelo Branco, Portugal
[3]INESC-ID, Instituto Superior Técnico, Technical University of Lisbon, Portugal
[4]Dept. of Telematics Engineering, Universitat Politécnica de Catalunya (UPC), Spain
{joao.dias, joao.isento, mario.pereira, bruno.silva}@it.ubi.pt, {vasco.g.soares, joeljr}@ieee.org,
{prbp, augusto.casaca}@inesc.pt, {cristina, javier.gallego}@entel.upc.edu

Abstract— **Vehicular Delay-Tolerant Networks (VDTNs) use vehicles to carry data between network nodes, enabling network connectivity in sparse or partitioned opportunistic networks, where the node density is not high enough to establish end-to-end links. By implementing a store-carry-and-forward paradigm, VDTNs allow delay-tolerant data traffic from a variety of vehicular applications to be routed over time, such as, but not limited to, information queries (e.g. weather reports, web browsing, business services, road conditions, traffic volume) and context- specific broadcasts (e.g. advertising, entertainment feeds). This paper presents a Web browsing application for VDTNs, called WWW@VDTN, and analyzes its performance in a laboratory VDTN testbed. Epidemic and Spray and Wait routing schemes were combined with different scheduling and dropping policies in the performance evaluation process. It is shown that WWW@VDTN works properly in the VDTN testbed. Thus, it is envisioned that a real VDTN will be able to support Web access. Furthermore, the performance studies also reveal that a combination of scheduling and dropping policies that takes into account information about the remaining lifetime of data, improves the performance of the routing schemes in terms of delivery probability and delivery delay.**

Index Terms— **Vehicular Delay-Tolerant Networks; World Wide Web; Browser; Performance Assessment; Testbed; WWW@VDTN**

I. Introduction

Vehicular delay-tolerant networks (VDTNs) [1] are a new approach for vehicular communications, which collects some contributions from DTNs [2] (store-carry-and-forward paradigm) and optical burst switching networks (data packets aggregation under network layer and out-of-band signaling). VDTNs proposal intend to improve the performance of data dissemination and routing in opportunistic vehicular networks characterized by highly dynamic topology, short contact durations, disruption, intermittent connectivity, significant loss rates, and frequent network partitions. In a VDTN, mobile and fixed nodes interact with each other to provide data dissemination and routing. Figure 1 illustrates these interactions among nodes. Terminal nodes are fixed devices and represent the access points to the VDTN network. Stationary relay nodes are located at road intersections and have store-and-forward capabilities. They have an essential role in low node density scenarios in improving the network performance [3]. Mobile nodes (e.g., vehicles) move on roads, collecting and disseminating data through the VDTN network.

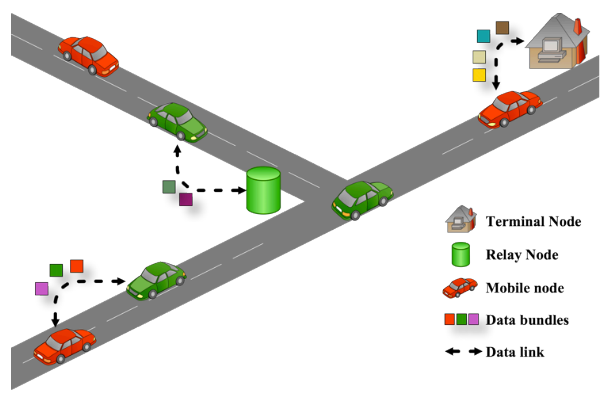

Fig. 1. Interactions between VDTN network nodes.

Road safety, traffic monitoring, driving assistance, entertainment, and delivering connectivity to rural/remote communities or catastrophe-hit areas are just a few examples of the many applications envisioned for VDTNs. They also may be used to support traditional Internet services, such as World Wide Web (WWW), electronic mail (eMail), or file transfer, but not on real time [4-5].

This paper proposes and demonstrates a WWW application for VDTNs, called WWW@VDTN. The performance of this application is evaluated in VDTN@Lab, a laboratory testbed created for assessing the VDTN architecture, protocols, services, and applications. Data traffic generated by this application is routed using two routing protocols (Epidemic [6] and Spray and Wait [7]), and different combinations of dropping and scheduling policies. The performance metrics considered were the percentage of successfully answered requests (e.g., delivery probability) and their average delivery delay.

The rest of this paper is organized as follows. Section II presents related work describing available vehicular testbeds or projects that emulate some Internet services. The design of the proposed Web browser application (WWW@VDTN) is presented on Section III. Section IV mentions the laboratory VDTN testbed where WWW@VDTN is executed and presents its performance assessment. Finally, Section V concludes the paper and points some directions for future work.

II. RELATED WORK

This section presents several projects and vehicular testbeds considered on the design of the proposed Web browsing application.

Cabernet [8] is a system that uses vehicles, equipped with IEEE 802.11 access points, to exchange data. This system supports applications for messages delivering (such as, eMail, traffic updates, parking information) or files (Web objects, documents, songs).

Thedu [9] is a system that allows users traveling in vehicles to use a Web search service. This system uses an Internet proxy to collect search results and pre-fetch result pages. When a vehicle has a contact opportunity, it downloads the responses for the user requests, if they are available.

CambodiaSchools.com [10] allows students in Cambodian schools located in remote rural areas, to request Web pages and use non-real time applications and Internet services like eMail and voice mail. Using buses or motorcycles equipped with mobile access points, data is collected and carried between hubs and spots with Internet access located at villages.

The Drive-thru Internet project [11] considers the use of IEEE 802.11 to allow users in moving vehicles to access to Internet services, such as eMail and file sharing tools based on the HTTP protocol. In this project, wireless LAN hot-spots, located at roadsides or in city areas like gas stations, restaurants and cafés, are exploited to allow data exchange with vehicles.

The proposal and construction of WWW@VDTN gathers contributions from the above-mentioned projects. The proposed application was deployed and demonstrated on a laboratory testbed and its performance evaluated.

III. WEB BROWSING APPLICATION FOR VDTNS

This section describes WWW@VDTN, a delay-tolerant Web browsing application proposed to offer WWW service on vehicular delay-tolerant networks. This section considers two subsections. The former presents the requirement analysis of WWW@VDTN, while the second describes the technologies used to create this application.

A. Requirement analysis

Before creating an application it is crucial to define all the steps and operations needed to implement the behavior that it is expected to perform. Unified modeling language (UML) was used to make a high-level design and requirements analysis of the WWW@VDTN. Figures 2 and 3 present the most important concepts of the application. It is important to notice that WWW@VDTN only runs on network nodes that perform traffic source and sink operations, not on relay nodes.

Figure 2 presents the use cases diagram of the WWW@VDTN client application. While the client application is running, a user may perform one of the following operations: *i*) *request functions*: including request a new Web page, save or reject a requested page; and *ii*) *statistical functions*: including access to performance metrics (e.g., number of requests sent by each source, requests' average delivery delay).

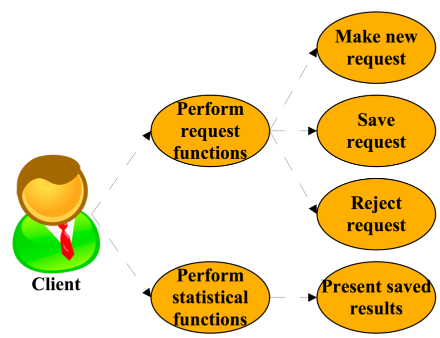

Fig. 2. UML use cases diagram of WWW@VDTN client application.

Figure 3 shows an activity diagram of the WWW@VDTN client illustrating the action when a user wants to request a new Web page. The user inserts the desired Uniform Resource Locator (URL) and submits his request in the client node (i.e., traffic source). This request is then packed and stored in a terminal node buffer while waiting for a transmission opportunity to a mobile node. When this contact occurs, this request is forwarded to the mobile node as part of a bundle. Then, VDTN network nodes like mobile nodes and relay nodes will store, carry and forward this bundle until it arrives at the server node (i.e., traffic source where requested Web page is stored). Each time the server node detects a new request, it tries to access the Internet and fetch the desired Web page. Then, a communication channel is opened (in the form of a stream) between the application and the remote Web server and all the relevant data is transferred. Finally, the data downloaded is aggregated in a bundle, which will be sent back to the user (i.e., the client node), as a response of the requested Web page. If the bundle time-to-live (TTL) expires while in transit in the VDTN network, the request/response is discarded.

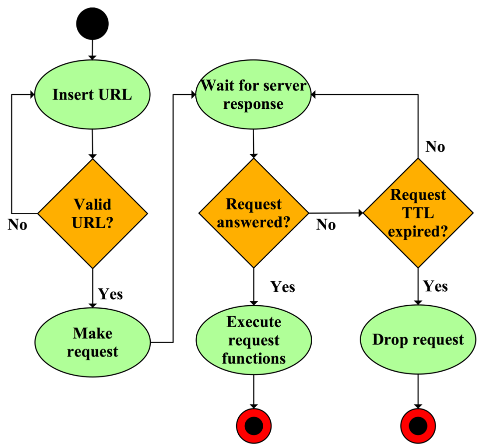

Fig. 3. UML activity diagram of WWW@VDTN client application.

B. *WWW@VDTN Design*

The proposed application considers two software modules (client and server). Both modules were created using the *C#* language and the *.Net Framework*, and run in desktops and laptops with the Windows 7 operating system.

The server module, in the testbed, provides logging information that allows following what actions are being performed by the server side application, as may be seen in Figure 4. For example, it is possible to view some information about senders (e.g., host name, address) and even about the requests (e.g., creation time, requested Web page).

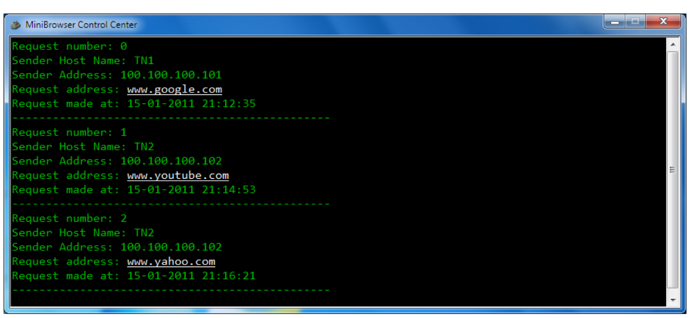

Fig. 4. Server module showing statistical information.

The client module is organized in three different areas: the textbox for inserting the URL, the retrieved Web pages list box and the page information box. Figure 5 illustrates the client module as it is presented to the user, performing the above-described functions. This module also allows the user to select a specific retrieved Web page and show it in the default browser installed in the computer.

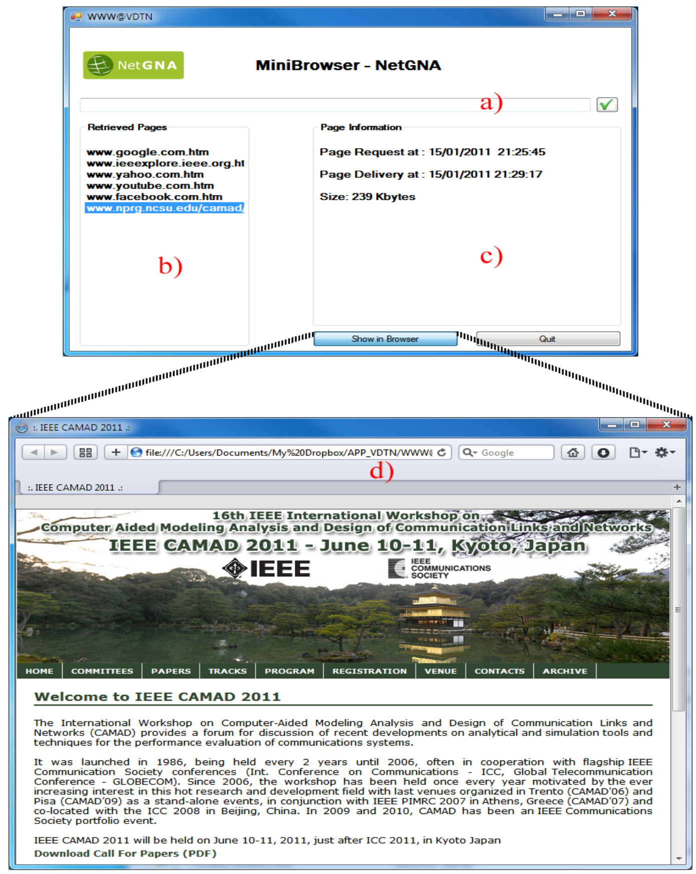

Fig. 5. User's interface of the WWW@VDTN client module, considering three areas: a) URL textbox, b) retrieved Web pages list box, c) Web page information box, and d) Web page displayed on the user default browser.

IV. Performance Assessment

This section briefly describes the VDTN laboratory testbed where the WWW@VDTN application is demonstrated and their performance assessed.

A. *VDTN@Lab Laboratory Testbed*

The VDTN@Lab laboratory testbed includes the three node types of VDTN networks (terminal, relay and mobile nodes). Terminal and relays nodes are emulated on desktop computers (iMacs - Intel(R) Core(TM) 2 Duo 2.66GHz + 4GB RAM), while HP Mini laptops (Intel(R) Atom 1.66GHz + 1GB RAM) coupled into LEGO MINDSTORMS NXT robotic cars, are used to emulate mobile nodes. All nodes use Bluetooth for out-of-band signaling and IEEE 802.11 connections for data transfers, and storage capacities to allow VDTN data communications.

The network scenario considered to evaluate the network performance of WWW@VDTN over VDTNs uses three terminal nodes, two relay nodes, and four mobile nodes. These mobile nodes move with different velocities. The fastest one is *Mobile node 1* and the slowest one is *Mobile node 4*. Of the remaining two nodes, the faster is *Mobile node 3*. Two terminal nodes run the client application acting as traffic sources and the third one acts as the traffic sink running the server application. Terminal nodes are located at the edge points of the laboratory. Two relay nodes are located at crossroads. Four mobile nodes move on roads between random positions and at different velocities, thus, simulating real vehicles behavior. All networks nodes assume different storage capabilities depending on their role. Mobile nodes have a 25MB buffer, relay nodes 75MB, and terminal nodes 50MB.

For the performance assessment, Web page requests are generated every 30 seconds by client terminal nodes (i.e., traffic sources). Each request has a time-to-live (TTL) of 15 minutes. The size of the requested Web pages varies in the range of [128 K, 1024 M] (Bytes). A Web page is dropped from the nodes' buffers when their TTL expires (and thus, they are no longer meaningful) or when congestion occurs. Each testbed experiments runs for one hour. Figure 6 shows the described testbed with all nodes and its interactions.

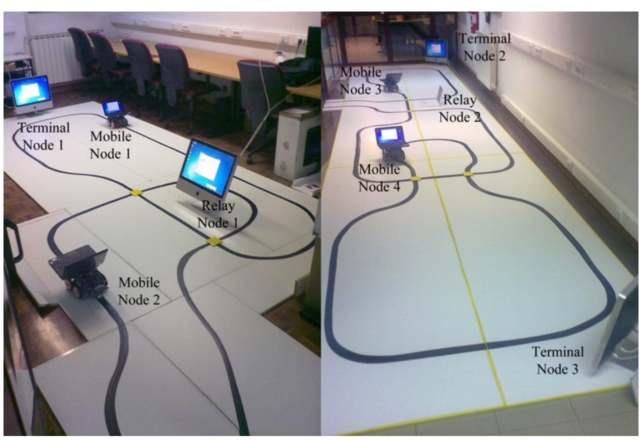

Fig. 6 – Photos of the VDTN laboratory testbed.

B. Performance Analysis

The performance assessment of the WWW@VDTN application over VDTNs assuming the use of Epidemic and Spray and Wait protocols as underlying routing schemes is presented in this sub-section. In addition, three different combinations of scheduling and dropping policies (FIFO, Remaining Lifetime (RL), and Random) described in [12] are enforced. Performance metrics considered in this study are the delivery probability (measured as the relation of the number of delivered Web pages requests out of the number of requested Web pages), and the average delivery delay (measured as the mean time between Web pages request and their successful delivery).

In order to increase the delivery probability and decrease their average delay, requests with longer remaining TTL are forwarded first while requests with smaller remaining TTL are dropped first. The reason why this is done is due to storage and bandwidth constraints, limiting the number of Web pages carried per node and the number of them exchanged at each contact opportunity.

Performance Analysis of Epidemic Routing Protocol

The performance analysis starts with a study of the delivery probability as a function of the Web page size, when Epidemic routing protocol is used. As may be seen in Figure 7, the Remaining Lifetime (RL) combination presents the best results. This combination presents gains of 3%, 4%, 4%, 5%, and 5% for the considered Web page response sizes, when compared with FIFO. It also improves the delivery probability approximately in 2%, 3%, 7%, 9%, and 9% when compared with Random policy.

Figure 7 shows that Remaining Lifetime policy also contributes to decrease the average delivery delay. When compared with FIFO, Web pages arrive to its final destination 32, 47, 92, 144, and 221 seconds sooner, and approximately 24, 35, 50, 70, and 85 seconds sooner when compared to Random policy.

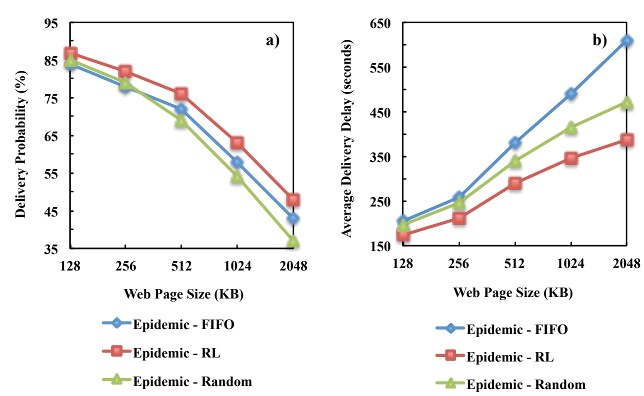

Fig. 7. a) Delivery probability and b) average delivery delay as function of the web page size, using Epidemic routing scheme with three different combinations of scheduling and dropping policies (FIFO, RL, and Random).

Performance Analysis of Spray and Wait Routing Protocol

Now, the study considers the performance assessment of VDTNs using binary variant of Spray and Wait routing mechanism, with *N*=3. Figure 8 shows the results observed for the delivery probability as a function of the Web page size. As may be seen, it confirms that a RL combination of scheduling and dropping policies increases the delivery probability in approximately 6%, 4%, 3%, 5%, and 6% when compared to FIFO, and in 4%, 3%, 6%, 9%, and 12% when compared with Random policy.

As expected, Figure 8 shows that RL also contributes to decrease the average delivery delay. When compared with FIFO, Web pages arrive to its final destination 12, 13, 50, 84, and 101 seconds sooner, and approximately 6, 8, 21, 32, and 49 seconds sooner when compared to Random.

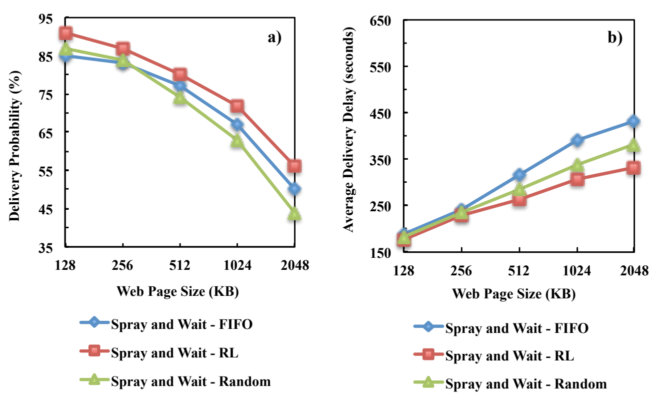

Fig. 8. a) Delivery probability and b) average delay as function of the Web page size using Spray and Wait routing scheme with three different combinations of scheduling and dropping policies (FIFO, RL, and Random).

V. CONCLUSIONS AND FUTURE WORK

This paper has presented a solution to provide WWW access over VDTNs, called WWW@VDTN. This Web browsing application has been demonstrated in a testbed and its performance assessment has been studied. It was shown that non-real time Web browsing can be deployed on VDTN networks. Different combinations of scheduling and dropping policies and two routing schemes (Epidemic and Spray and Wait) has been used. It has been shown that a combination of scheduling and dropping policy that uses a remaining lifetime criteria outperforms the commonly used FIFO and Random combinations. For the considered network scenario, Spray and Wait has a better performance when compared to Epidemic routing protocol.

The performance of VDTNs may be improved with new routing protocols and, consequently, VDTN applications such as WWW@VDTN, will benefit of their use as well. Furthermore, fragmentation/de-fragmentation issues may be considered at the core network while aggregation/de-aggregation algorithms may improve the network performance at the edge of the network. These topics may be addressed in future works.

ACKNOWLEDGMENTS

Part of this work has been supported by the *Instituto de Telecomunicações*, Next Generation Networks and Applications Group (NetGNA), Portugal, in the framework of the VDTN@Lab Project, and by the Euro-NF Network of Excellence of the Seventh Framework Programme of EU, in the framework of the Specific Joint Research Project VDTN.

REFERENCES

[1] V. N. G. J. Soares, F. Farahmand, and J. J. P. C. Rodrigues, "A Layered Architecture for Vehicular Delay-Tolerant Networks," in *IEEE Symposium on Computers and Communications* (ISCC 2009), Sousse, Tunisia, July 5-9, 2009, pp. 122 - 127.

[2] V. Cerf, S. Burleigh, A. Hooke, L. Torgerson, R. Durst, K. Scott, K. Fall, and H. Weiss, *"Delay-Tolerant Networking Architecture,"* in *RFC 4838*, 2007.

[3] J. J. P. C. Rodrigues, V. N. G. J. Soares, and F. Farahmand, *"Stationary Relay Nodes Deployment on Vehicular Opportunistic Networks,"* in *Mobile Opportunistic Networks: Architectures, Protocols and Applications*, M. K. Denko, Ed. USA: Auerbach Publications, CRC Press, 2011.

[4] Y. Khaleda, M. Tsukadaa, J. Santab, J. Choia, and T. Ernst, "A Usage Oriented Analysis of Vehicular Networks: From Technologies to Applications," *Journal of Communications, Academy Publisher*, vol. 4, no. 5, pp. 357-368, June 2009.

[5] Y. Toor, P. Muhlethaler, A. Laouiti, and A. D. L. Fortelle, "Vehicle Ad Hoc Networks: Applications and Related Technical Issues," *IEEE Communications Surveys & Tutorials*, vol. 10, no. 3, pp. 74-88, 2008.

[6] A. Vahdat and D. Becker, *"Epidemic Routing for Partially- Connected Ad Hoc Networks,"* Duke Univ., Tech. Report, April 2000.

[7] T. Spyropoulos *et al.*, *"Spray and Wait: An Efficient Routing Scheme for Intermittently Connected Mobile Networks,"* in *ACM SIGCOMM 2005*, Philadelphia, PA, USA, 2005, pp. 252 – 259.

[8] J. Eriksson, H. Balakrishnan, and S. Madden, *"Cabernet: Vehicular Content Delivery Using WiFi,"* in *14th ACM International Conference on Mobile Computing and Networking (MobiCom 2008)*, San Francisco, California, USA, September 14-19, 2008, pp. 199-210.

[9] A. Balasubramanian *et al.*, *"Web Search From a Bus,"* in *ACM MobiCom - The Second Workshop on Challenged Networks (CHANTS 2007)*, Montreal, Canada, Sept. 9-14, 2007, pp. 59 - 66.

[10] A. Pentland *et al.*, *"DakNet: Rethinking Connectivity in Developing Nations,"* in *IEEE Computer*, vol. 37, 2004, pp. 78-83.

[11] Jörg Ott, Dirk Kutscher, *"Applying DTN to Mobile Internet Access: An Experiment with HTTP,"* Technical Report TR-TZI-050701, Technologiezentrum Informatik, Universität Bremen; July 2005.

[12] J. A. Dias, J. N. Isento, V. N. G. J. Soares, and J. J. P. C. Rodrigues, *"Impact of Scheduling and Dropping policies on the Performance of Vehicular Delay-Tolerant Networks,"* in *IEEE ICC 2011*, Kyoto, Japan, June 05-09, 2011.