Performance Implications of Fragmentation Mechanisms on Vehicular Delay-Tolerant Networks

João A. Dias[1], João N. Isento[1], Joel J. P. C. Rodrigues[1], Paulo Rogério Pereira[2], and Jaime Lloret[3]
[1] Instituto de Telecomunicações, University of Beira Interior, Portugal
[2] INESC-ID, Instituto Superior Técnico, Technical University of Lisbon
[3] Universidad Politécnica de Valencia, Spain

{joao.dias; joao.isento}@it.ubi.pt, joeljr@ieee.org, prbp@inesc.pt, jlloret@dcom.upv.es

Abstract — Vehicular Delay-Tolerant Network (VDTN) is a new disruptive network architecture where vehicles act as the communication infrastructure, furnishing low-cost asynchronous opportunistic communications, variable delays and bandwidth limitations defining a non-TCP/IP (Transmission Control Protocol/Internet Protocol) network. A VDTN assumes asynchronous, bundle-oriented communication, and a store-carry-and-forward routing paradigm. VDTNs should make the best use of the tight resources available in the network. In order to optimize the data exchanged among nodes at contact opportunities, increasing the bundle delivery and improving the overall network performance, fragmentation mechanisms are used. This paper presents several fragmentation techniques (proactive, source, reactive, and toilet paper) for VDTNs and evaluates their performance through a laboratory testbed. It was shown that both reactive fragmentation approaches (reactive and toilet paper) perform slightly better than proactive fragmentation approaches (proactive and source) and non-fragmentation approaches.

Keywords — Vehicular Delay-Tolerant Networks; Fragmentation; Performance Evaluation; Prototype.

I. INTRODUCTION

Vehicular Delay-Tolerant Networks (VDTNs) [1] are a kind of vehicular networks where a variety of mobile and fixed nodes interact with each other in order to support communications in sparse, remote, or disconnected scenarios. VDTNs may have three different types of nodes: terminal, relay, and mobile nodes. Terminal nodes are fixed devices placed at the edge of the network, and represent the access points to the VDTN. Mobile nodes (e.g. vehicles) are exploited to collect and disseminate data bundles through the network. They move along roads, carrying data that must be delivered to terminal nodes. Mobile nodes may also act as terminal nodes, generating and receiving data. Finally, stationary relay nodes are fixed devices located at crossroads. These nodes interact directly with mobile nodes, that can deposit and pickup data. Relay nodes are used to increase the number of contact opportunities in scenarios with low node density. Hence, as the number of contact opportunities increases, the number of bundles delivered also increases leading to a decrease of the bundles delivery delay [2].

The VDTN architecture differs from Delay-Tolerant Networks (DTNs) by placing the bundle layer under the network layer introducing an IP over VDTN approach. It also performs out-of-band signaling, with control and data planes separation [1]. The protocol data unit at the VDTN bundle layer is called *bundle*, which aggregates datagrams based on the used assembling algorithm. VDTNs implement the same store-carry-and-forward paradigm proposed for DTNs [3]. This paradigm solves the problem caused by disconnection, long delays and intermittency. Figure 1 illustrates this paradigm as well as the interactions between VDTN network nodes.

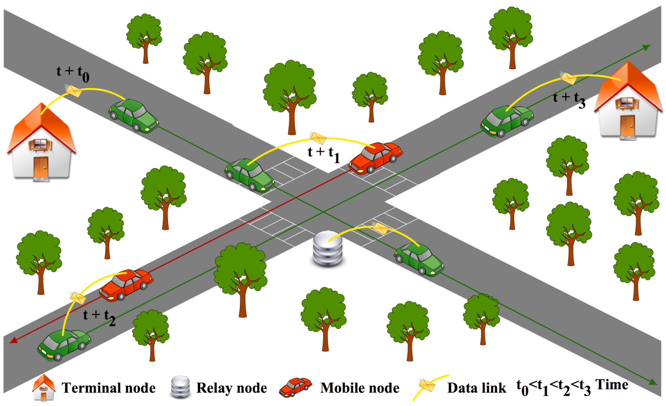

Fig. 1 - Illustration of the store-carry-and-forward paradigm for VDTNs and the interactions between the three types of VDTN network nodes.

Although VDTNs present some differences when compared to DTNs, they face the same challenges when it comes to establishing network connectivity between vehicles. Most of these challenges are due to the high mobility and velocity of vehicles, which cause constant changes in the network topology and short contact durations. The contact duration plays an important role in the study of fragmentation approaches for VDTNs. When two nodes are in contact among them, they have a limited time to exchange bundles. If the contact time expires and it is under a bundle transfer, this bundle transmission will be incomplete. If it no fragmentation approach is considered, the incomplete bundle will be deleted. Deleting an incomplete bundle, results in a waste of resources that leads to a decrease of the bundle delivery probability and an increase of bundle delivery delay. In order to maximize network resources and to increase the bundle delivery ratio, this paper presents and analyses the performance of the following fragmentation approaches for VDTNs: *proactive*, *source*, *reactive*, and *toilet paper*.

The remainder of the paper is organized as follows. Section II presents related work focusing on DTN fragmentation approaches while Section III presents and describes the proposed fragmentation approaches. Section IV discusses the results of the experiments and Section V provides conclusions and directions for future work.

II. Related Work

This section surveys the most relevant related literature about fragmentation, ranging from fragmentation in IP networks to fragmentation in vehicular networks. Fragmentation in IP networks occurs when an IP datagram is larger then the Maximum Transmission Unit (MTU) of the underlying data link technology. The datagram is divided into pieces that can be reassembled later using several fields (IP Source, Destination, Identification, Total Length, and Fragment Offset) and flags ("More Fragments" and "Don't Fragment") presented on the IP header. The mechanisms of IP fragmentation and reassembly are reported in [4], and are illustrated on Figure 2.

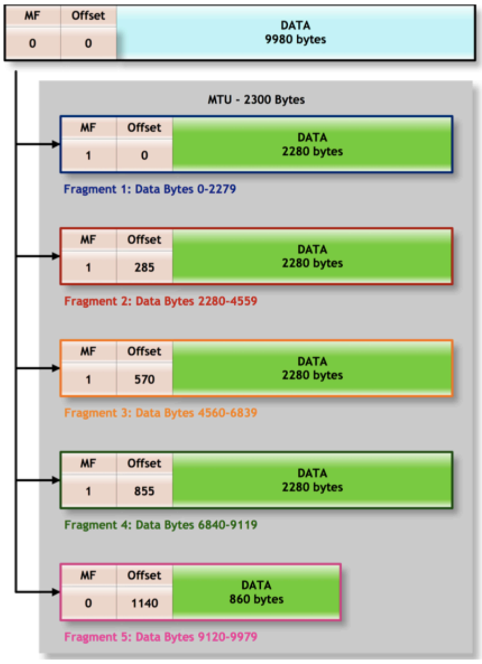

Fig. 2 - Illustration of fragmentation in IP networks.

Although allowed, IP fragmentation is discouraged and is considered harmful. It can lead to performance loss or to a complete communication failure that increases the packet loss probability [5]. To avoid fragmentation on IP networks, transmitters must determine the MTU for a specific path, in a dynamic way, sending multiple packets of different sizes with the IP header "*don't fragment*" flag active. Another possible solution to solve this problem is allowing transmitters to chose a MTU based upon conservative expectations on the operating environment and the application demands.

Vehicular networks are characterized by intermittent connectivity representing a lack of an end-to-end path. Taking into account these constraints, each contact opportunity must be used to exchange as much bundles as possible. Each contact opportunity has a reduced time and represents a limitation for bundle forwarding. These restrictions motivate the use of fragmentation on vehicular communications. Another factor that can motivate fragmentation is buffer space. Assuming that nodes have limited storage capacity it may happen that in a contact opportunity a node does not have space available to receive the entire bundle. In this case, and in order to optimize each contact opportunity, messages should be fragmented.

A new Media Access Control (MAC) protocol for wireless Local Area Networks (LANs) based on dynamic fragmentation is proposed [6]. This protocol adapts the fragment size based on channel condition information obtained from preceding fragment transmission.

In [7] an adaptive fragmentation scheme for VANETs is presented. This approach relies on wireless channel time varying property and on the VANETs network load condition.

The DTN architecture supports two types of fragmentation: proactive and reactive. Under proactive fragmentation, a node with previous knowledge of the available link or with information about buffer conditions on the next hop, splits bundles into fragments before transmitting them. Any network node may perform this type of fragmentation, but only the final destination nodes are responsible for extracting the fragmented bundles and reassembling them into the original bundles. A special case of proactive fragmentation, called source fragmentation, is presented in [8]. In this approach, the source node splits the original bundle into n non-overlapping fragments with similar size. Thus, source node forwards these fragments sequentially. Intermediary nodes forward the fragments unchanged because no further fragmentation is allowed.

In reactive fragmentation, bundles are divided when a transmission is interrupted by a sudden link failure. In this case, the intervenient nodes reconcile the segments not transmitted yet. The bundle segments already transmitted are aggregated into valid bundles, agreeing on how much data was successfully forwarded to avoid losses. A variant of reactive fragmentation may also be considered. Under it, the size of resulting fragments is not arbitrary, but is defined by the originator. This variant is called the *toilet paper* approach [9].

III. Fragmentation in VDTNs

In a VDTN network, the high mobility and velocity of nodes leads to constant changes in the network topology and short contact durations. When two nodes establish contact they start to perform the control plane operations to select which bundles should be exchanged among them. Afterwards, at the data plane, a sender node forwards the selected bundles (at the control plane) until the contact time expires or the connection is broken. If the contact is interrupted when a bundle is under transmission, the sender assumes that bundle was successfully transmitted, while the receiver only has an incomplete bundle. If no fragmentation mechanisms are considered, this incomplete bundle is deleted resulting on a waste of network resources. Deleting an incomplete bundle might result in a possible retransmission of the bundle increasing the bundle delivery delay. With this increase of the bundle delivery delay, bundles may be dropped from buffer nodes if their time-to-live (TTL) expires, resulting in the bundle delivery ratio decreasing and decreasing also the performance of VDTNs.

In order to improve the performance of VDTNs and overcome the above-presented problem, a fragmentation approach is proposed, based on several fragmentation mechanisms for DTNs available in the literature (*proactive*, *proactive source*, *reactive*, and *toilet paper*) [8]. In the *proactive* mechanism, bundles are fragmented when the size of data bundles to be transmitted is larger than the data allowed for a contact. To perform this type of fragmentation in VDTNs, both nodes in contact perform the control plane operations in order to select which bundles they should exchange. At the control plane, the fragmentation module determines which bundles will be completely transmitted and which bundle should be fragmented, as well as the size of the fragmented bundle. The bundle fragment will be treated as an independent bundle for buffer management and routing decisions. At the end of the control plane both nodes know the amount of data that will be transmitted, whether they are complete bundles or fragments of bundles. Only the destination node can reassembled a bundle if all of its fragments are received. If a fragment does not reach the destination, the bundle is lost. Fragments may be lost due to TTL that may expire or due to buffer congestion. Figure 3 illustrates the above-presented mechanism.

A variant of proactive fragmentation, called *source* fragmentation scheme, is studied. The main difference to *proactive* fragmentation is that bundles are divided into fragments with a given size. This process is performed by the traffic sources when bundles are generated. After the traffic *source* fragmentation, no further fragmentation takes place.

In the *reactive* mechanism, bundles are fragmented in real time (on the fly), when nodes are exchanging bundles among them (using the data plane). To perform such operation, both nodes in contact should be able to determine which part of the bundle has been successfully transferred and which part not. Afterwards, the receiver creates a fragment using the data that has been successfully received, while the sender creates a fragment with the remaining data. Figure 4 illustrates the operations of the fragmentation module when the reactive fragmentation is performed.

Finally, a variant of *reactive* fragmentation, called *toilet paper* is also considered. In this variant of *reactive* fragmentation the size of the resulting fragments is not arbitrary, but limited to a given size defined by the originator.

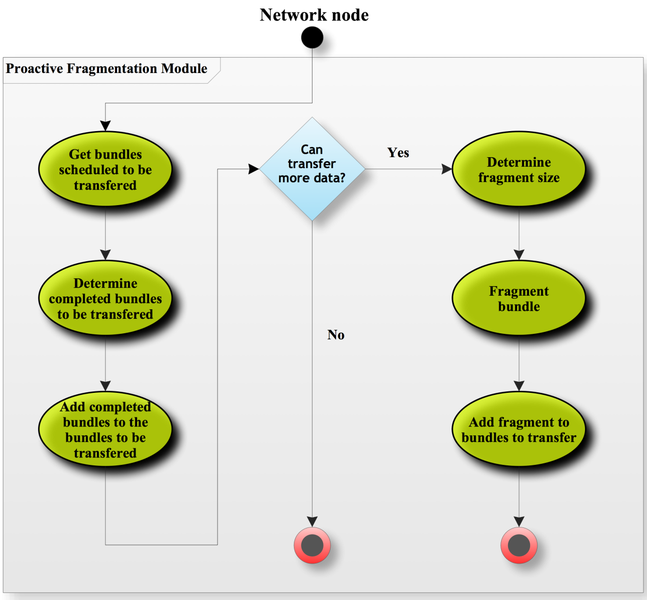

Fig. 3 – Fragmentation module operations when the proactive fragmentation is performed.

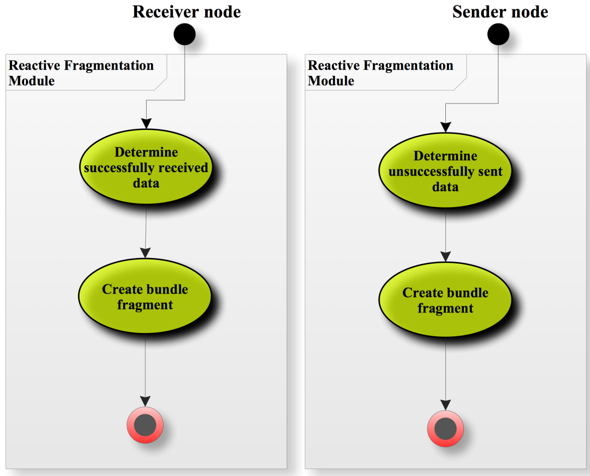

Fig. 4 – Fragmentation module operations when the reactive fragmentation is performed.

IV. VDTN@LAB LABORATORY TESTBED

The VDTN@Lab, a VDTN prototype available at laboratorial environment, allows the validation and performance evaluation of VDTN services, applications, and protocols. This testbed uses desktops, laptops, netbooks, and robotic vehicles. Laptops are used to emulate terminal and relay nodes. Mobile nodes are emulated by netbooks coupled into LEGO MINDSTORMS NXT robots. These netbooks have networking and storage capabilities. LEGO MINDSTORMS NXT robots are programmed for having a random movement across roads. All the testbed nodes are equipped with Bluetooth and IEEE 802.11b/g devices in order to perform the out-of-band signaling with the separation of control and data planes. Bluetooth is used to exchange signaling information (control plane), while the IEEE 802.11g is used to exchange data bundles. A set of software modules were created using C# programming language and .Net Framework. These software modules run in laptops and netbooks with Microsoft Windows 7 operating system, and provide management tools and advanced statistics reports.

For the performance evaluation of fragmentation mechanisms presented in this paper, a scenario with four mobile nodes, two relay nodes, and three terminal nodes was considered. Mobile nodes have a random movement across paths in order to simulate random routes and moves at different speeds. Paralleling with a study based on a testbed composed by real vehicles [10], and assuming a scale of 1:50 (1m in the laboratory testbed represents 50m in a real scenario), mobile nodes move with speeds of 48 Km/h, 40 Km/h, 36 Km/h, and 24 Km/h. Mobile nodes have a buffer of about 25MB. Relay nodes are placed at crossroads and have a buffer with a capacity of 75MB. Terminal nodes are located at different places of the laboratory and have a buffer with 50MB of capacity. Figure 5 shows photos of the VDTN@Lab testbed and all the above-described interactions and behaviors.

Data bundles are generated at each 20 seconds and are sent from a random source node to a random destination node. The bundle size is uniformly distributed in the range of [256 Kbytes, 4096 Kbytes], and its TTL is fixed at 20 minutes. Bundles are deleted from buffers when their TTL expires or congestion occurs. Experiments were conducted assuming a FIFO-HeadDrop combination of scheduling and dropping policies [11].

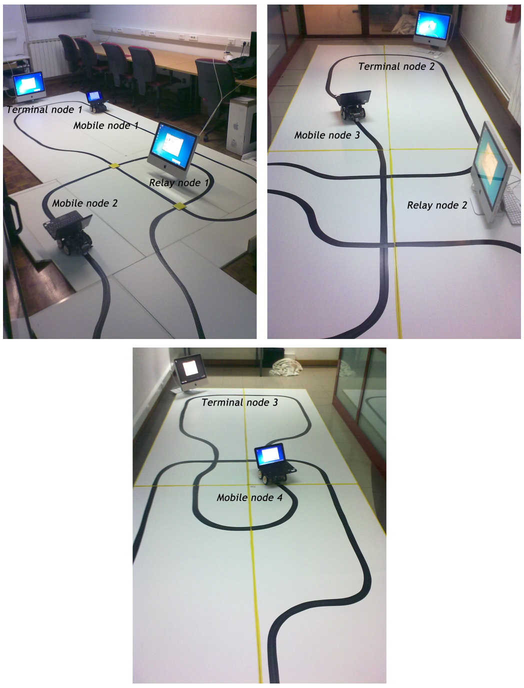

Fig. 5 – Photos of the VDTN@Lab testbed scenario.

V. Performance Analysis

In order to analyze the impact of the above-described fragmentation approaches on the performance of a VDTN network, several experiments on VDTN@Lab were conducted. For each result, an average of 30 experiments was considered. The *proactive*, *proactive source* (using 3 equal size fragments), *reactive*, and *toilet paper* (with 128 KB fragments) fragmentation mechanisms were considered.

Performance metrics used in this study are the bundle delivery probability and the bundle average delay. The bundle delivery probability is measured as the relation between the number of unique delivered bundles and the number of forwarded bundles. The bundle average delay is measured as the time between bundles creation and delivery.

This section includes three sub-sections, considering the Epidemic [12], Spray and Wait [13], and PRoPHET [14] routing protocols, respectively.

A. *Performance Analysis for the Epidemic routing protocol*

This analyses starts with the performance study about the impact of different fragmentation mechanisms in the behavior of the Epidemic routing protocol. Figure 6 shows the results observed for the bundle delivery probability. As may be seen, both types of *reactive* fragmentation perform slightly better when compared to non-fragmentation case, while proactive (*source*) fragmentation always perform worse. This happens because bundles are fragmented at the time of their creation, meaning that all the bundle fragments have to reach its final destination in order to be reassembled the original bundle. On Epidemic protocol, given the buffer congestion caused by it, the, most part of its fragments will be dropped.

When compared to non-fragmentation, *toilet paper* approach presents gains of 3%, 4%, 9%, 8%, and 15% (for bundles size equals to 256, 512, 1024, 2048, and 4096 Kbytes, respectively). The *reactive* approach presents gains of 1%, 3%, 6%, 5%, and 17% when compared to non-fragmentation case. The performance of the *proactive* approach is very similar to the *reactive* approach. However, it performs slightly worse than that, decreasing the bundle delivery probability in 1%, 2%, 2%, 1%, and 3%.

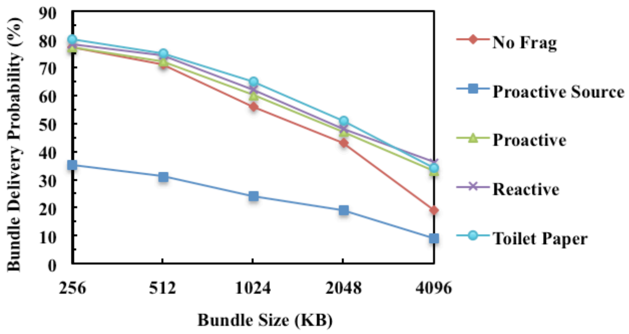

Fig. 6 – Bundle delivery probability as function of bundle size for the Epidemic routing protocol.

Figure 7 shows that with the introduction of *reactive* and *proactive* fragmentation mechanisms, bundles arrive to their final destination later than without fragmentation, for bundles with size less than 2048 Kbytes. As may also be seen in this figure, when the bundles size increase, the *reactive* and *proactive* fragmentation mechanisms tend to deliver bundles sooner than the non-fragmentation approach. In this case, only *proactive source* fragmentation delivers bundles a slightly later than the non-fragmentation approach.

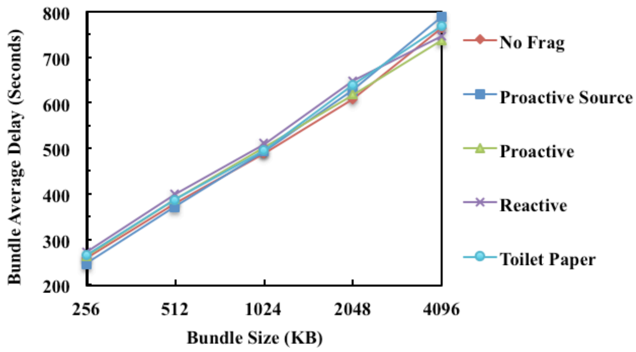

Fig. 7 – Bundle average delay as function of bundle size for the Epidemic routing protocol.

B. *Performance Analysis for Spray and Wait routing protocol*

This sub-section analyses the performance of fragmentation mechanisms when the Binary Spray and Wait routing protocol is considered. As may be seen in Figure 8, both *reactive* fragmentation mechanisms perform better than the non-fragmentation mechanism. When compared to the non-fragmentation, *toilet paper* presents gains of 1%, 2%, 8%, 11%, and 20% (for bundles size equals to 256, 512, 1024, 2048, and 4096 Kbytes, respectively). The *reactive* mechanism presents gains of 2%, 1%, 7%, 9%, and 20% when compared to the non-fragmentation approach. The *proactive* mechanism presents gains of 1%, 1%, 4%, 7%, and 17%, when compared to the non-fragmentation approach. The *proactive source* fragmentation always performs worse than the non-fragmentation approach.

Figure 9 presents the results obtained for the bundle average delay. All mechanisms presents similar results, although the both *reactive* and *proactive* fragmentation mechanisms delivers bundles slightly sooner than the non-fragmentation approach. Increasing the bundles size, *proactive source* mechanisms tends to deliver bundles later than all other mechanisms.

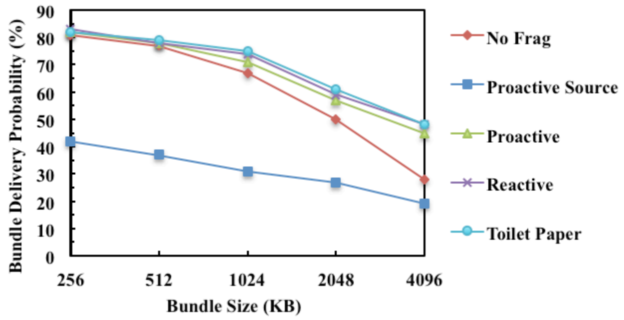

Fig. 8 – Bundle delivery probability as function of bundle size for the Spray and Wait routing protocol.

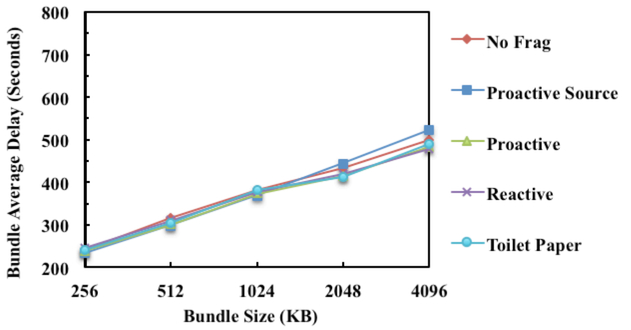

Fig. 9 – Bundle average delay as function of bundle size for the Spray and Wait routing protocol.

C. *Performance Analysis for PRoPHET routing protocol*

Finally, the performance of fragmentation mechanisms is studied when the PRoPHET routing protocol is enforced. Figure 10 shows the results observed for the bundle delivery probability. As may be seen, both *reactive* fragmentation mechanisms perform better than the non-fragmentation approach. When compared to the non-fragmentation approach, *toilet paper* presents gains of 1%, 10%, 11%, and 11% (for bundles size equals to 512, 1024, 2048, and 4096 Kbytes, respectively). The *reactive* approach presents gains of 2%, 8%, 6%, and 13% when compared to the non-fragmentation case. *Proactive* fragmentation also presents gains when compared to the non-fragmentation mechanism. It performs 1%, 5%, 4%, and 9% better than this. The *proactive source* fragmentation always performs worse than the non-fragmentation approach. It is observed that fragmentation only provide a significant gain with the increase of bundles size. This happens manly due to the contact duration. As bundles are getting bigger and the contact duration is maintained, less complete bundles will be transmitted. Without a fragmentation mechanism this will result on a waste of network resources.

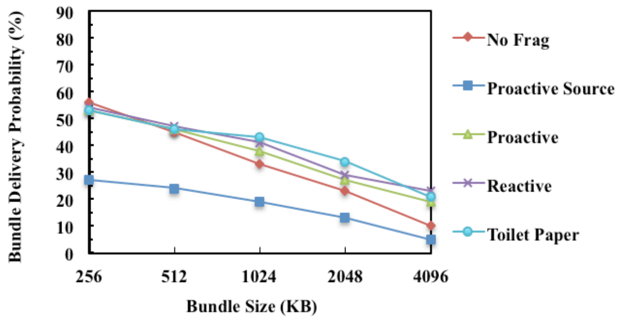

Fig. 10 – Bundle delivery probability as function of bundle size for the PRoPHET routing protocol.

Figure 11 addresses the bundle average delay for the same routing scheme. As may be seen, all the studied fragmentation mechanisms present similar performance, although the *reactive* approach delivers bundles sooner as bundles size increase.

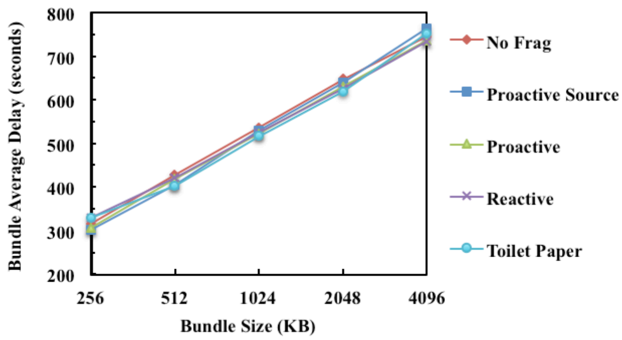

Fig. 11 – Bundle average delay as function of bundle size for the PRoPHET routing protocol for the studies fragmentation mechanisms.

VI. Conclusions and Future Work

This paper presented several fragmentation mechanisms for VDTNs, founded on DTN-based approaches (*proactive*, *source*, *reactive*, and *toilet paper*). The influence of different fragmentation mechanisms on the performance of VDTNs was analyzed. The study was conducted through a laboratory testbed, called VDTN@Lab. In this context, it was observed that both reactive fragmentation approaches maximize the bundle delivery probability, while the bundle average delay remains practically the same for all approaches.

The carried experiments were performed enforcing the above-mentioned fragmentation mechanisms on the Epidemic, Spray and Wait, and PRoPHET routing protocols. In terms of bundle delivery probability, Spray and Wait performs better than the remaining protocols for all the considered fragmentation approaches. This happens because Spray and Wait limits the number of copies in the network. This means that buffers congestion is less likely to occur, or occurs later in each testbed experiment. Because of this, bundles will be dropped mainly due to TTL expiration and not by buffer congestion, allowing more bundles or fragments to reach its final destination.

This study may be extended with the introduction of more routing protocols, such as MaxProp. Network management may also be included for further works, as well as real deployment of VDTNs.

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 Project VDTN@Lab, by the Euro-NF Network of Excellence of the Seventh Framework Programme of EU, in the framework of the Specific Joint Research Project VDTN, and by National Funding from the FCT – *Fundação para a Ciência e a Tecnologia* through the PEst-OE/EEI/LA0008/2011 Project.

REFERENCES

[1] V. N. G. J. Soares, F. Farahmand, and J. J. P. C. Rodrigues, "A Layered Architecture for Vehicular Delay-Tolerant Networks," in *The Fourteenth IEEE Symposium on Computers and Communications* (ISCC 2009), Sousse, Tunisia, July 5-8, 2009, pp. 122-127.

[2] 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.), Auerbach Publications, CRC Press, February, 2011.

[3] V. Cerf, S. Burleigh, A. Hooke, L. Torgerson, R. Durst, K. Scott, K. Fall, and H. Weiss, "Delay-Tolerant Networking Architecture," in RFC 4838, April, 2007, [Online] Available: http://www.rfc-editor.org/rfc/rfc4838.txt.

[4] Jon Postel, "Internet Protocol," in RFC 791, September, 1981, [Online] Available: http://www.ietf.org/rfc/rfc791.txt.

[5] Christopher A. Kent and Jeffrey C. Moguk, "Fragmentation considered harmful," SIGCOMM Computer Communication Review, vol. 25, no. 1, pp. 75-87, January 1995.

[6] Kim B.-S., Y. Fang, T.F. Wong, and Y. Kwon, "Throughput Enhancement Through Dynamic Fragmentation in Wireless LANs," IEEE Transactions on Vehicular Technology, vol. 54, no. 4, pp. 1415-1425, July 2005.

[7] H. Liu, H. Sheng, Z. Lv, L. Li, and C. Ma, "A cross layer design of fragmentation and priority scheduling in vehicular ad hoc networks," in *7th World Congress on Intelligent Control and Automation* (WCICA 2008), June 25-25, 2008, pp. 6157-6160.

[8] Mikko P., Ari K., and J. Ott, "Message Fragmentation in Opportunistic DTNs," in *9th IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks* (WOWMOM 2008), Newport Beach, CA, USA, 23-26 June, 2008.

[9] S. Farrell, S. Symington, and H. Weiss, "Delay-Tolerant Networking Security Overview," Internet Draft, March 2009, [Online] Available: http://tools.ietf.org/html/draft-irtf-dtnrg-sec-overview-06.

[10] M. Rubinstein *et al*, "Measuring the Capacity of In-Car to In-Car Vehicular Networks", *IEEE Communications Magazine*, vol. 47, no. 11, pp. 128-136, November 2009.

[11] João A. Dias, João N. Isento, Vasco N. G. J. Soares, and Joel J. P. C. Rodrigues, "Impact of Scheduling and Dropping Policies on the Performance of Vehicular Delay-Tolerant Networks," in the *IEEE International Conference on Communications (ICC 2011)*, Kyoto, Japan, June 5-9, 2011.

[12] A. Vahdat and D. Becker, "Epidemic Routing for Partially-Connected Ad Hoc Networks," Duke University, Technical Report, CS-2000-06, April 2000.

[13] T. Spyropoulos, K. Psounis, and C. S. Raghavendra, "Spray and Wait: An Efficient Routing Scheme for Intermittently Connected Mobile Networks," in *ACM SIGCOMM 2005 - Workshop on Delay Tolerant Networking and Related Networks* (WDTN-05), Philadelphia, PA, USA, August 22-26, 2005, pp. 252-259.

[14] A. Lindgren, A. Doria, E. Davies, and S. Grasic, "Probabilistic Routing Protocol for Intermittently Connected Networks," Internet Draft, July 12, 2010, [Online] Available: http://tools.ietf.org/html/draft-irtf-dtnrg-prophet-06.