SLOPE: A System for rapid Deployment of VANET Communication Protocols

Sascha Schnaufer* Holger Füßler* Matthias Transier*
Wolfgang Effelsberg*

1 Introduction

Vehicular Ad-Hoc Networks (or VANETs) are a special kind of Mobile Ad-Hoc Network (or MANET), where wireless-equipped (road) vehicles form a network with no additional infrastructure. While many communication scenarios exist for these networks, government-sponsored research activity like the German Network-on-Wheels project mainly focuses on their application to increase vehicular safety with extra room for application increasing driver convenience.

The quest for a communication system to alleviate active vehicular safety has lead to a break with many conventional paradigms of protocol architectures finding relatedness as much in traditional wireless IP networks as in sensor networks with quite different types of network protocols. [1] extensively discusses challenges in this area and states the fact that a network stack in such a safety-critical system, where the main influence on its usability is the quality of service of the underlying network protocol, can hardly be viewed as separable as in IP networks. On the contrary we think it is very likely that in VANETs, highly specialized protocols with a minimal packet sending requirement will be deployed, since one application's packet might stop another application from delivering a vital message.

However, this necessity to employ special cross-layer protocols creates a fundamental dilemma: While on one hand communication people like us are unable to create and tune vehicular safety applications, vehicular safety engineers are mostly unaware of the consequences the sending of a packet might have on this or other applications' functioning. At this point communication engineers could focus on optimizing link-layer reliability or medium access and leave multi-hop communication to the application layer neglecting the fact that ad-hoc networking protocols are very sensitive and even very simple ones might not work at all [2].

Right now this dilemma leads to a stall in the development and deployment of VANET safety protocols. In this paper we present a solution that alleviates the problems with a system approach. SLOPE (Self-Organizing Communication with Protocol Elements) provides a simple, yet powerful method to use so-called protocol elements which are in fact cross-layer communication protocols encapsulated in a software environment requiring a minimal effort from the application developers for filling out some code. In classical terms of stack architectures, these protocol elements are mostly based on link-layer data packets to communicate with direct neighbors but based on high-level application specific

*Sascha Schnaufer, Holger Füßler, Matthias Transier, and Wolfgang Effelsberg are with 'Computer Science IV' at the University of Mannheim.

information when it comes to the rest of the network.

In this sense, they provide multi-hop functionality, but rather on the basis of information than of an unchanged end-to-end packet payload. Of course, we can only treat a packet payload as information if the semantics of the encoded data is clear. Thus, applications play an important role within the SLOPE proposal.

In Sec. 2 we will briefly outline some architectural aspects of SLOPE. Sec. 3 will give an overview of what we would like to show with the poster. For a more elaborate introduction to the necessity of SLOPE and some of its aspects, please check [3].

2 What is SLOPE?

In its core, SLOPE is (a) a way of dividing network protocol functionality different to standard protocol layering and (b) providing the software engineering and a runtime environment to alleviate protocol development. From a protocol's perspective, the result is a layer network-to-application protocol fulfilling a special application task and encapsulating VANET intelligence. From a user's perspective, the protocol is divided into two parts, so-called protocol elements (or PEs). One part—the lower part as it would be in classical layers—is provided by SLOPE while the other is contributed by the safety application developer. The key idea is that this VANET know-how is hidden from the application developer. On the other side, the application allows to classify packet similarity on the basis of the contained information. This enables the lower PE to suppress redundant packets or—to be more precise, to choose the level of redundancy required to match the application's needs.

To allow a rapid protocol and development and deployment, the SLOPE system is implemented in Java with some system-dependent parts for low-level network access. Exploiting the full power of Java, we provide the application interface by means of object-oriented Java interfaces, allowing a separation of the application and the communication system. In addition, we provide some powerful mechanisms like, e.g., quasi-automatic packet marshaling to further alleviate protocol development. In the end, building an application protocol should be as easy as extending a Java class.

3 The Poster

On the poster, we will show both the protocol and the software architecture of the SLOPE system and its runtime environment. In addition, we will outline an exemplary protocol element from both the perspective of an application protocol developer and the one from a communication system designer.

Ackowledgements

Sascha Schnaufer acknowledges the support of the German Ministry for Education and Research (BMB+F) for the "Network-on-Wheels" project under contract no. 01AK064F.

References

[1] H. Füßler, M. Torrent-Moreno, M. Transier, A. Festag, and H. Hartenstein. Thoughts on a Protocol Architecture for Vehicular Ad-Hoc Networks. In *Proc. of WIT 2005*, pages 41–45, Hamburg, Germany, March 2005.

[2] S.-Y. Ni, T. Yu-Chee, C. Yuh-Shyan, and S. Jang-Ping. The Broadcast Storm Problem in a Mobile Ad Hoc Network. In *Proc. of ACM MobiCom '99*, pages 151–162, Seattle, Washington, August 1999.

[3] S. Schnaufer, H. Füßler, M. Transier, and W. Effelsberg. Vehicular Ad-Hoc Networks: Single-Hop Broadcast is not enough. In *Proc. of WIT 2006*, pages 49–54, Hamburg, Germany, March 2006, to appear.