Master Thesis
Emulation of automotive networking protocols for the Environment for Generic In-vehicle Networking Experiments

Topic Description

Today's In-Vehicular Networks (IVN) are in the midst of a significant paradigm shift. The low-bitrate CAN and FlexRay are being rapidly replaced by high-bandwidth Ethernet-based solutions using the support of the IEEE 802.1Q [1] family of Time-Sensitive Networking (TSN) standards. To ensure high reliability, flexibility, and deterministic packet delivery within these new In-Vehicular systems, researchers and manufacturers require tools that allow for rapid-prototyping and simultaneously produce results that correspond to those of real-world deployments. One of such tools is the Environment for Generic In-vehicular Networking Experiments [2], otherwise called EnGINE. The framework provides a flexible environment for repeatable, reproducible, and autonomous Ethernet-based IVN experiments which are performed using open-source solutions on commercial off-the-shelf hardware. Solutions used include Linux and OpenVSwitch, as well as the Intel I210 NICs.

The switch towards ethernet in IVNs won't happen instantaneously. While we are seeing more Ethernet in IVNs, today's cars continue to use mainly CAN or FlexRay as the networks interconnecting multiple vehicle components. However, with the increasing requirements on those networks in terms of available bandwidth, we see manufacturers switching towards Ethernet at least for the backbone of the IVN. For a period, both legacy and ethernet technologies will be coexisting. Thus, to accurately represent current functionality of IVNs in EnGINE, we require an ability for emulation of CAN/LIN/FlexRay traffic that can be injected via an emulated gateway into the IVN.

The goal of this thesis is to implement and evaluate methods for emulation of CAN and/or FlexRay traffic that can be injected on to the ethernet TSN backbone network provided by the EnGINE framework. There are multiple ways of achieving the desired outcome. Firstly, any available packet traces from real CAN/FlexRay deployments could be used. The challenge within this approach concerns how to realize the gateway from e.g. CAN to Ethernet and back. An outcome of this solution could be, e.g., preparation of repayable Ethernet packet traces with certain types of traffic and their analysis within the network.

Another option is to simulate the CAN/FlexRay traffic using the OMNeT++ discrete event simulator [3,4]. To achieve this, the FiCo4OMNeT [5,6] framework would need to be used. The realization could be done as real-time simulation and injection of the traffic onto the network towards a source, or generation of packet traces that can then be replayed on the network. Simulation of the traffic would be desirable as it would not limit the traffic patterns to pre-defined ones. Other methods/ideas are welcome.

The outcome of the thesis should include prepared traces or method(s) for real-time CAN/FlexRay to Ethernet traffic emulation that's integrated into the EnGINE framework as a stack/application. Furthermore, it should include an evaluation of end-to-end delay within the proposed emulation solution.

(See next page for more details)

Your tasks

1. Define and investigate available methods for emulation of such traffic within the IVN
2. Familiarize with EnGINE framework and its Ansible configuration syntax
3. Search for and analyze available CAN/FlexRay packet traces and CAN/FlexRay to Ethernet gateway functionality
4. Familiarization with OMNeT++ [3,4] and FiCo4OMNeT [5,6]
5. Prepare ways of emulating the CAN/FlexRay traffic
6. Execute and evaluate of experiments that include traffic generated using the prepared methods of CAN/FlexRay emulation
7. Execute and evaluate some experiments that include a traffic matrix corresponding to a real-world vehicle

Required Experience

- General knowledge on computer networking
- Knowledge of IEEE 802.1Q TSN standards is a plus
- Knowledge of CAN/LIN/FlexRay is a plus
- Some experience with discrete event network simulation or other ways of emulation
- Experience with OMNeT++ is a plus

Additional Information

- Offered as a Master Thesis, IDP or a Bachelor Thesis could be considered
- May be extended into a publication upon completion

References

[1] *"IEEE Standard for Local and Metropolitan Area Network--Bridges and Bridged Networks,"* in IEEE Std 802.1Q-2018 (Revision of IEEE Std 802.1Q-2014) , vol., no., pp.1-1993, 6 July 2018, doi: 10.1109/IEEESTD.2018.8403927.

[2] Filip Rezabek, Marcin Bosk, Thomas Paul, Kilian Holzinger, Sebastian Gallenmüller, Angela Gonzalez, Abdoul Kane et al. "EnGINE: Developing a Flexible Research Infrastructure for Reliable and Scalable Intra-Vehicular TSN Networks."

[3] András Varga, and Rudolf Hornig. "An overview of the OMNeT++ simulation environment." In *Proceedings of the 1st international conference on Simulation tools and techniques for communications, networks and systems & workshops*, pp. 1-10. 2008.

[4] Levente Mészáros, Andras Varga, and Michael Kirsche. *"Inet framework."* In *Recent Advances in Network Simulation*, pp. 55-106. Springer, Cham, 2019.

[5] Buschmann, Stefan, Till Steinbach, Franz Korf, and Thomas C. Schmidt. "Simulation based timing analysis of FlexRay communication at system level." In *SimuTools*, pp. 285-290. 2013.

[6] https://github.com/CoRE-RG/FiCo4OMNeT

Contact/Supervisors

Marcin Bosk bosk@in.tum.de