Poster Abstract: HiL meets Commodity Hardware – SimbaR for coupling IEEE 802.11 Radio Channels

Mario Franke[†] and Florian Klingler[*†]
[*]Dept. of Computer Science, [†]Paderborn University, Germany
https://www.fklingler.net

***Abstract*—We present Simulation-based Radio (SimbaR), an extension to our open-source prototyping system LAN Radio to couple IEEE 802.11-based communication channels of real world and simulation using commodity hardware. These coupled radio channels enable testing of prototypes (e.g., vehicular ECUs) in large scale simulation studies without the need for changing the IEEE 802.11 access layers (i.e., MAC and PHY) of these devices. However, fairness for channel access has not been investigated for such systems, yet. By applying MAC layer adjustments to the testbed at runtime, SimbaR can control the fairness for channel access between simulated stations and real world prototypes (e.g., an ECU). Besides transceiving information from simulation to the real world and vice versa, SimbaR can recreate interference observed in the simulation in the real world. In first experiments we show the effectiveness of our open-source prototyping approach by highlighting the necessity of proper channel access schemes and interference generation for coupled radio channels.**

I. Introduction

Vehicular Networking research evolved from pure simulative studies to deployed prototypes and integrated Hardware-in-the-Loop (HiL) approaches which couple large scale simulation scenarios with a physical Device Under Test (DUT) [1]–[3]. Further, concepts have been presented to couple IEEE 802.11 radio channels of simulation and real world to field test Electronic Control Units (ECUs) in large scale simulations in real time [4], [5]. In particular, LAN Radio [4] uses commodity hardware to couple radio channels by transmitting IEEE 802.11 WLAN frames from within the simulation to the real world and vice versa. It allows to integrate physical devices, e.g., ECUs of cars, without changing their access layers (e.g., MAC and PHY) for communication.

However, this coupling of wireless channels introduces a set of research challenges, which to the best of our knowledge have not yet been sufficiently addressed in literature. By investigating the communication performance, e.g., the goodput of a simulated and a physical station, it can be observed that communication delay due to coupling leads to unfair channel access, especially in situations where interference within the simulation is not represented accordingly in the real world.

In this paper, we present Simulation-based Radio (SimbaR)[1] which addresses this shortcoming by controlling the MAC of commodity hardware. SimbaR bidirectionally couples communication between simulated stations and DUTs and recreates interference on the physical radio channel based on the used simulation models.

[1]https://www.fklingler.net/software/

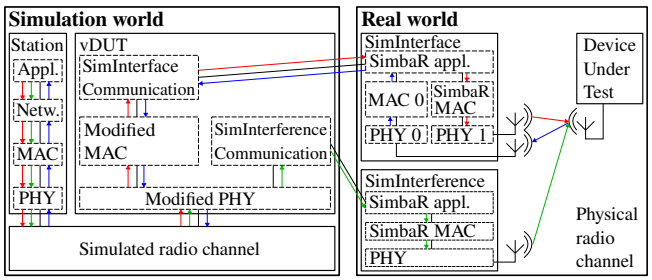

Figure 1. Architecture of SimbaR for connecting simulation world with real world. Although only a single simulated station is visualized, the SimbaR concepts supports multiple stations.

II. SimbaR Architecture

In Figure 1 we show the architecture of SimbaR which builds upon concepts published in [4], [5]. We start with an event-based wireless network simulation scenario (consisting of a number of simulated stations) and add a node called virtual Device Under Test (vDUT) which acts as the simulative representative of the physical DUT. Further, our system connects the simulation via a network connection to two hardware devices (SimInterface, and SimInterference) which are responsible to transceive data frames from simulation and real world and to generate interference in the real world, respectively. For our hardware testbed we build upon the GNU/Linux OpenWrt operating system framework targeting embedded devices.

Simulation World: From a simulation perspective, the vDUT has to accomplish two tasks: *(a)* forwarding packets from simulated radio channel to the physical radio channel (cf. *red and green* path in Figure 1), and, *(b)* transmitting packets sent by the DUT via the simulated radio channel. Packets received by the SimInterface are integrated in the simulation (cf. *blue* path in Figure 1) with the help of a concurrent thread. This thread autonomously creates events and inserts them into the Future Event Queue (FEQ) of the event-based simulator.

Real World: The Simulation Interface (SimInterface) is responsible for data exchange on the physical radio channel. We use two radios, one for receiving and one for transmitting data frames from/to the DUT. For the transmitting radio we disable the WLAN DCF and CSMA/CA behavior similar to what has been used in [6] to avoid double contention of channel resources. Packets which already got access to the simulated radio channel shall not compete a second time for getting access to the physical radio channel. Using Radiotap headers

40th IEEE International Conference on Computer Communications (INFOCOM 2021)

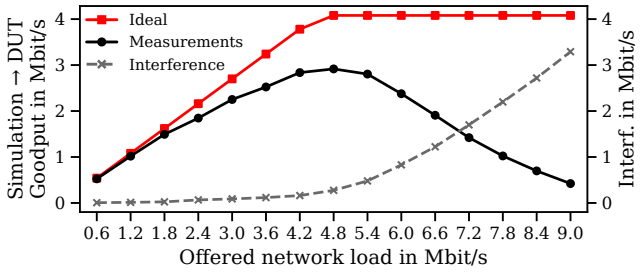

Figure 2. Measured goodput from simulation to DUT is compared to an ideal baseline. The dashed line shows the generated interference in the real world.

we perform packet injection such that each frame is transmitted according to its meta data, e.g., transmit power. Similar to the vDUT and with the help of a second radio, the SimInterface can receive and transmit at the same time by exploiting the capture effect [7]. If the DUT's transmitted packets are received with high enough power, they can be decoded even there might be interference or a transmission of the SimInterface itself.

In order to represent observed interference from the simulation in the real world (cf. *green* path in Figure 1), the Simulation Interference (SimInterference) device creates interference on the physical radio channel, again with disabled DCF and CSMA/CA. However, frames transmitted by the SimInterference device intentionally have a wrong Frame Check Sequence (FCS) in their MAC header such that these packets will be discarded by the DUT. We pay special attention to multiple interfering packets overlapping in the time domain by avoiding interference frames to be sequentially transmitted on the physical channel.

III. Performance Evaluation

To show the effectiveness of SimbaR, we perform a wireless network simulation incorporating one physical DUT and nine simulated stations. For both hardware devices (SimInterface and the SimInterference) we use PC Engines APU2 single board computers equipped with MikroTik R11e-5HnD WLAN cards able to operate in the 5.9 GHz band at 10 MHz channel bandwidth as required by IEEE 802.11p. As simulation framework we use Veins 5.0 and configure the wireless interfaces according to the same specification. For the sake of simplicity and being able to compare the results to baseline analytical calculations, we do not consider mobility and arrange all stations to be within each other's communication range.

As prime performance metric we consider the overall goodput between the SimInterface, representing all nine simulated stations, and the physical DUT. This metric allows us to get insights into the channel fairness under the consideration of interference. To perform these measurements, we use a computer-controlled DUT which exactly behaves like simulated stations with regards to the application layer. All stations, including the DUT, periodically transmit 400 Byte packets at QPSK-$^1/_2$ (6 Mbit/s). In Figure 2 we show the results from these experiments by comparing the measured received goodput at the DUT to an ideal baseline. Further, we also show the rate (in Mbit/s) of generated interference in the real world.

For very low offered loads, the measured goodput is very similar to the ideal goodput due to little or no interference. However, with increasing offered network load, the generated interference increases due to packet collisions in the simulation. We measure a local maximum at around 4.8 Mbit/s where a channel busy ratio of around 68 % can be observed (data not shown due to space constraints). The rationale here is clear: If we would not recreate interference from the simulation in the physical world, the DUT would experience with increasing offered network loads an even emptier channel. Consequently, the DUT would then be able to transmit more frames towards the simulation which would create unfair channel behaviour. Since the DUT is not able to transmit and receive at the same time, and thus frames transmitted by SimInterface while the DUT was transmitting are lost, the measured goodput is lower than the ideal goodput.

IV. Conclusion

For HiL simulations incorporating DUTs which require communication, e.g., in the vehicular networking domain, coupling wireless radio channels between simulation and real world is needed. Current approaches in the literature considering commodity hardware for coupling WLAN based radio channels in a HiL context only focus on exchange of information without appropriately modelling interference in the real world. In this paper we fill this gap by presenting our open-source prototyping platform SimbaR. Our testbed allows to have detailed control over the physical radio channel by adapting the MAC behavior of commodity hardware at runtime. First results show the necessity of modelling interference to gain fair channel access for coupling simulation and real world.

Acknowledgements

Supported partly by the EU/EFRE.NRW Hy-Nets4all project.

References

[1] M. Eisenbarth, M. Wegener, R. Scheer, J. Andert, D. S. Buse, F. Klingler, C. Sommer, F. Dressler, P. Reinold, and R. Gries, "Towards Smart V2X-Connected Powertrains," *IEEE Vehicular Technology Magazine (VTMag)*, 2020.

[2] J. Manco, G. Baños, Guillermo, J. Härri, and M. Sepulcre, "Prototyping V2X applications in large-scale scenarios using OpenAirInterface," in *2020 IEEE Vehicular Networking Conference (VNC 2020)*, IEEE, Dec. 2020.

[3] G. Shah, M. Saifuddin, Y. P. Fallah, and S. Datta Gupta, "RVE-CV2X: A Scalable Emulation Framework for Real-Time Evaluation of CV2X-based Connected Vehicle Applications," in *2020 IEEE Vehicular Networking Conference (VNC 2020)*, IEEE, Dec. 2020.

[4] F. Klingler, G. S. Pannu, C. Sommer, and F. Dressler, "Connecting Simulation and Real World: IEEE 802.11p in the Loop," in *23rd ACM International Conference on Mobile Computing and Networking (MobiCom 2017), Poster Session*, Snowbird, UT: ACM, Oct. 2017, pp. 561–563.

[5] C. Obermaier, R. Riebl, and C. Facchi, "Fully Reactive Hardware-in-the-Loop Simulation for VANET Devices," in *21st IEEE International Conference on Intelligent Transportation Systems (ITSC 2018)*, Maui, HI: IEEE, Nov. 2018, pp. 3755–3760.

[6] M. Vanhoef and F. Piessens, "Advanced Wi-Fi Attacks Using Commodity Hardware," in *30th Annual Computer Security Applications Conference (ACSAC 2014)*, New Orleans, LA: ACM, Dec. 2014, pp. 256–265.

[7] P. Fuxjaeger and S. Ruehrup, "Validation of the NS-3 Interference Model for IEEE802.11 Networks," in *8th IFIP Wireless and Mobile Networking Conference (WMNC 2015)*, Munich, Germany: IEEE, Oct. 2015, pp. 216–222.

EUROPÄISCHE UNION
Investition in unsere Zukunft
Europäischer Fonds
für regionale Entwicklung

EFRE.NRW
Investitionen in Wachstum
und Beschäftigung