A New Standard: Redesigning Autonomous Vehicle Communication Using Multi-Party Computation

Research Question: How can MPC be used to improve autonomous vehicle communication?

Written by:
Sever Latysov s.latysov@student.tudelft.nl
Supervised by:
Zekeriya Erkin z.erkin@tudelft.nl

TU Delft Delft University of Technology

1. Autonomous Vehicle Communication

Vehicle-to-vehicle (V2V) communication enables autonomous vehicles in range of each other to establish a network connection to share data [1]. Communication between vehicles is useful as it is much safer if a vehicle knows what other vehicles are doing instead of predicting what the other vehicles are doing based on sensor input. Up to 80 percent of all unimpaired crashes' scenarios could potentially be addressed by V2V [2].

2. Multi-Party Computation

Multi-party computation (MPC) is a cryptographic technique that allows a set of parties to compute the output of a function while not revealing their input data to the other parties [3].

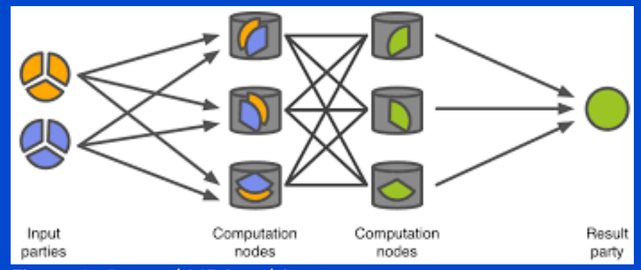

Figure 1: General MPC architecture

3. Methodology

Two methods were employed during this research. The first method was to perform a literature study and the second was to conduct interviews with experts in the field of autonomous vehicle communication and MPC to gain an understanding of the problems and considerations in the development of vehicle communication systems and to discuss the feasibility of our proposed solutions.

4. MPC Architectures

Cloud-Aided Two-Party Computation (CA-2PC): Two vehicles in range of each other perform a two-party computation based on garbled circuits. Both parties are assisted by cloud servers which are used for preprocessing. Information between the vehicles is transmitted using V2V communication. The structure is illustrated in Figure 2.

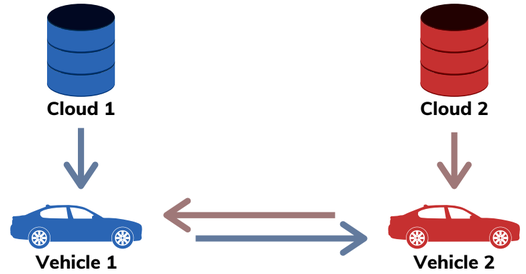

Figure 2: Proposed cloud-aided two-party computation architecture.

Three-Party Computation (3PC): Two vehicles distribute their data among the computational parties using additive secret sharing. The computational parties consist of three servers, two of which are from autonomous vehicle companies and the third from a government institution for traffic management. The output of the computation is returned to the vehicles. The structure is illustrated in Figure 3.

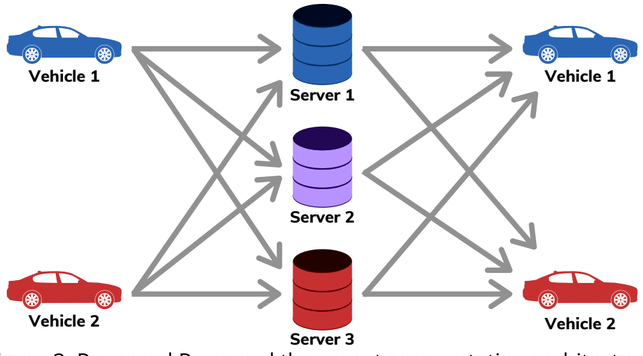

Figure 3: Proposed Proposed three-party computation architecture.

5. Discussion

Two different architectures were proposed because they both appear to be feasible whilst having different benefits and shortcomings.

CA-2PC requires no intermediate parties and therefore acts more independently, which is valuable for a self-driving car, but the vehicles communicate through V2V and communication can get disrupted because of the rapidly changing node topology. 3PC is much more reliable, as the returning output can be received even after the vehicles separate, but 3PC depends on commercial companies working together.

The autonomous vehicle industry is at the moment not concerned with improving autonomous vehicle communication. It has been sidelined until the safety of self-driving cars is fully guaranteed. For MPC to be used in the industry more interest in vehicle communication needs to develop first.

6. Conclusion

Using MPC vehicle communication can be made secure and privacy-preserving. This allows autonomous vehicles to share more data, achieve better situational awareness and thereby make the roads safer and more environmentally friendly.

Further experimentation is required to determine if real-time MPC for the application in V2V communication is possible. Additionally, further research is needed to prove the feasibility of both architectures.

Sources

Figure 1 - https://edps.europa.eu/sites/edp/files/publication/17-06-09_triin-siil_sharemind_en.pdf
[1] - Steven E Shladover. Connected and automated vehicle systems: Introduction and overview. Journal of Intelligent Transportation Systems, 22(3):190–200, 2018.
[2] - Siva RK Narla. The evolution of connected vehicle technology: From smart drivers to smart cars to. . . selfdriving cars. Ite Journal, 83(7):22–26, 2013.
[3] - Yehuda Lindell. Secure multiparty computation (mpc). IACR Cryptol. ePrint Arch., 2020:300, 2020.