This thesis presents a robust, flexible and a real-time architecture for Software-in-the-Loop testing of connected vehicle safety applications. Emerging connected and automated vehicles (CAV) use sensing, communication and computing technologies in the design of a host of new safety applications. Testing and verification of these applications is a major concern for the automotive industry. The CAV safety applications work by sharing their state and movement information over wireless communication links. Vehicular communication has fueled the development of various Cooperative Vehicle Safety (CVS) applications.

Development of safety applications for CAV requires testing in many different scenarios. However, the recreation of test scenarios for evaluating safety applications is a very challenging task. This is mainly due to the randomness in communication, difficulty in recreating vehicle movements precisely, and safety concerns for certain scenarios. We propose to develop a standalone Remote Vehicle Emulator (RVE) that can reproduce V2V messages of remote vehicles from simulations or from previous tests, while also emulating the over the air behavior of multiple communicating nodes. This is expected to significantly accelerate the development cycle. RVE is a unique and easily configurable emulation cum simulation setup to allow Software in the Loop (SIL) testing of connected vehicle applications in a realistic and safe manner. It will help in tailoring numerous test scenarios, expediting algorithm development and validation as well as increase the probability of finding failure modes. This, in turn, will help improve quality of safety applications while saving testing time and reducing cost.

The RVE architecture consists of two modules, the Mobility Generator, and the Communication emulator. Both of these modules consist of a sequence of events that are handled based on the type of testing to be carried out. The communication emulator simulates the behavior of MAC layer while also considering the channel model to increase the probability of successful transmission. It then produces over the air messages that resemble the output of multiple nodes transmitting, including corrupted messages due to the collision. The algorithm that goes inside the emulator has been optimized so as to minimize the communication latency and make this a realistic and real-time safety testing tool. Finally, we provide a multi-metric experimental evaluation wherein we verified the simulation results with an identically configured NS3 simulator. With the aim to improve the quality of testing of CVS applications, this unique architecture would serve as a fundamental design for the future of CVS application testing.