University of Colorado, Boulder

Network Systems

ECEN/CSCI 5273

Final Report

V2V Broadcasting Protocol

Project Customers

Name: Jose Santos Address: University of Colorado, Boulder Phone: (303)735-0102 Email: Jose.Santos@colorado.edu

Team Members

Name: Isha Burange Phone: (720) 447 1511 Email: isbu5273@colorado.edu	Name: Bhoomika Singla Phone: (720) 313 4957 Email: bhsi9770@colorado.edu
Name: Kiran Jojare Phone: (720) 645 6212 Email: kijo7257@colorado.edu	Name: Kevin Jones Phone: (503) 867 7438 Email: kejo7187@colorado.edu

Table of Contents:

Problem and Need

About 29% of accidents taking place on the highways are rear-end collisions. This serves as a use case for collision avoidance with the help of v2v (vehicle to vehicle) wireless communication. In this work, we propose a network architecture in which the vehicles communicate with each other. The goal of the project is to successfully send warning messages to the vehicles to provide the drivers sufficient time to slow down to avoid a chain reaction of rear-end collisions. For the scope of this project, we are assuming multiple vehicles moving in a single lane. Each vehicle is treated as a node with an unique ID which will have its own proximity table. Proximity table acts like a register which registers each and every vehicle in a particular vehicle's vicinity. Whenever a collision happens, the impacted vehicle sends an alert message to the rear vehicles and waits for an ack from those registered in its proximity table.

Our aim is to construct a network design for the communication between different vehicles on highways in order to avoid rear-end collisions. The idea is to connect all the vehicles on the same LAN network. Whenever a crash occurs, an alert message (**E**mergency **W**arning **M**essage) is broadcasted to the other vehicles. This EWM provides an alert to the following vehicles about the mis-hap and helps them in giving them time for deceleration. Once the vehicle in front receives acks from rear vehicles it stops sending the broadcast messages.

For a given node, the proximity table will keep track of the nodes within the proximity range and will be kept updated on a periodic basis so as to avoid network bandwidth wastage and eliminate redundant EWMs. The table update plays a crucial role as it avoids waiting for ACKs by other vehicles and hence the proximity table update will be performed in parallel with the transmission of EWMs.

Previous Work

“V2V Wireless Communication Protocol for Rear-End Collision Avoidance on Highways” by Fei Ye, Matthew Adams and Sumit Roy discussed four main challenges while implementing v2v wireless connection for rear-end collision avoidance. They proposed an integrated protocol design and evaluation for rear-end collision avoidance on highways. The paper also mentions rear end collision avoidance protocol which satisfies the stringent delay constraint. Both single lane and multi-lane scenarios are simulated. Their work emphasizes on sending an alert message to the following vehicles which will give them enough of Available Maneuvering Time to avoid the collision.

“Using vehicle-to-vehicle communication to Improve Traffic Safety in Sand-Dust Environment” by Jinhua Tan, Xuqian Qin and Li Gong conducted a study of a car-following model. This car-following model was inspired by Newell’s model which was developed in 1961. Newell’s model considered velocity-headway relation in real life traffic. The paper is a case-study of V2V wireless communication in Sand-Dust environment where the authors have discussed two main types of perturbations caused by a sudden stop. They have used this data to investigate multiple-vehicle collisions.

“Vehicle-to-Vehicle Safety Messaging in DSRC” by Qing XU, Raja Sengupta, Tony Mak and Jeff KO discusses the feasibility of sending safety messages from vehicle to vehicle in the DSRC(Dedicated Short-Range Communications) control channel. The authors propose to use TCP as it is a reliable transmission protocol and ensures that each byte of the message is received with certainty. The DSRC safety messages can be sent over LAN. They have modeled a single-hop local-area communication service delivering messages within their lifetime.

There are also many applications available for crash avoidance such as Intersection Movement Assist, Left Turn Assist, Emergency Electronic Brake Light, Forward Collision Warning, Blind Spot Warning, Do-not-stop Warning.

Specific Objectives

Level 1

We aim to build a v2v wireless communication protocol based on location broadcasting to avoid collisions between vehicles. As part of the communication protocol, a vehicle will send an emergency warning message (EWM) to the rear vehicles based on the proximity range. For level 1 we are considering vehicles running in a single lane and stuck in a traffic congestion, such that the relative positions of the vehicles are fixed with respect to each other. The deployment and simulation for this will be done in the Network systems lab in CU Boulder.

Level 2

As part of this level, we aim to extend the idea from level 1 to single-lane dynamic motion of vehicles, where we will keep updating the proximity table every 1 second, based on the nodes falling within the proximity range. The table update is necessary to avoid waiting for ACKs from non-relevant nodes and to ensure the integrity of collision avoidance. These updates will be done concurrently to the transmission of EWMs.

Level 3

Extending the level 2 to handle collisions in case of multi-lanes (3 lanes), which will be done as part of the future scope of the project. In case of multi-lane collision handling, we'll pass the lane ID along with message lifetime as part of the broadcast message to ensure that vehicles can disregard irrelevant packets(i.e messages from other lane vehicles). Since latency of the warning message is critical to collision avoidance, we analyze the latency of the EWMs and compare it with the traditional scenarios.

Functional Requirements

Block Diagrams

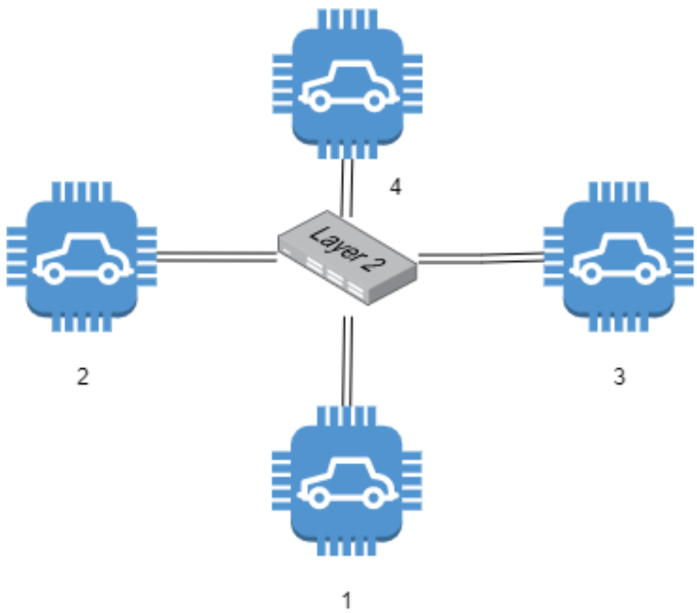


Figure 1: A simple model for V2V OTA communications

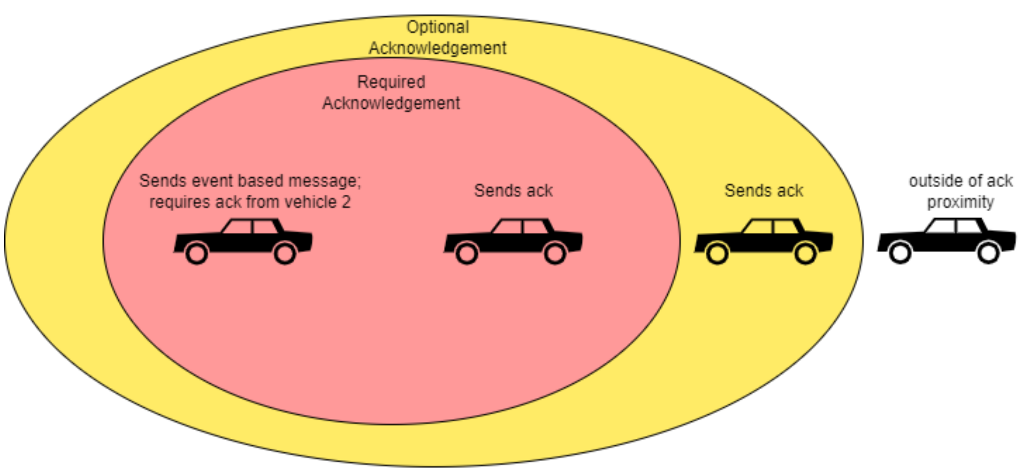


Figure 2: Spatially dependent acknowledgement system

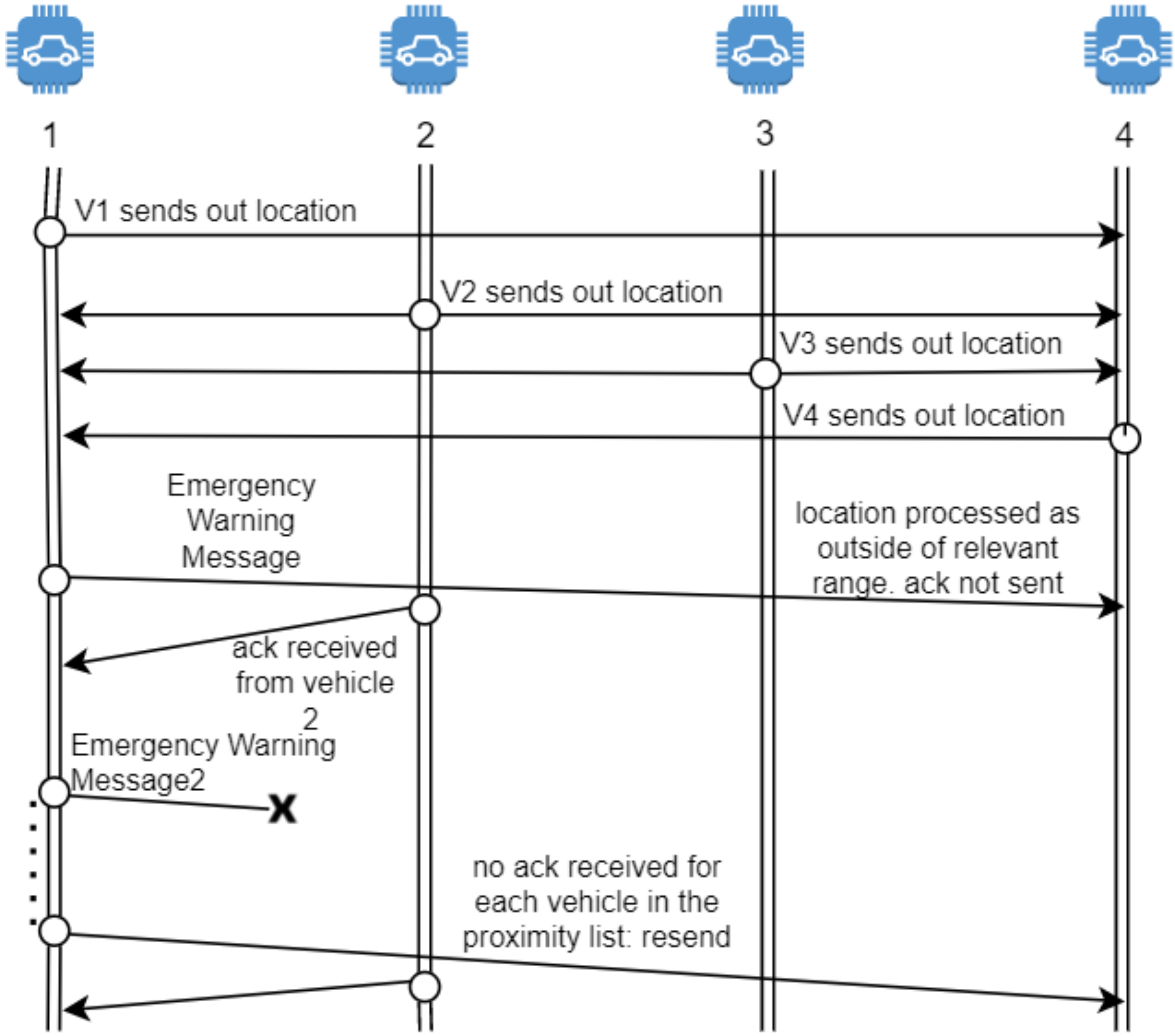


Figure 3: Example v2v call flow

Functional Description

The objective of this project is to build a network optimized for v2v communication based on a simplified scenario. This will be accomplished by implementing a custom protocol over the broadcast channel.

Several servers will be connected over the same LAN to simulate OTA transmission. The servers will broadcast an ID, time, and location. Other servers will listen and record servers within their proximity. Given an emergency event, braking scenario etc, a server will broadcast the event. Corresponding servers within the proximity will then record the event, and send an acknowledgement. If no acknowledgements are received from the servers in the proximity list, the original message will be resent. This process will continue for a time where the event is still relevant.

An optional implementation of the spatially based acknowledgement system is to vary the distance requirements used for the proximity table, and the distance requirement for sending an acknowledgment. This gives the table time to update given a vehicle falls out of the required acknowledgement area (see figure 3).

In order to analyze the networking system in various scenarios, logs will be used to record activity and characterize the communication latencies against other technology implementations.

Critical Project Elements

Technical C.P.E.

CPE 1.1: Implementation of vehicle ad-hoc network using broadcast oriented and location based routing

The majority of vehicle-to-vehicle network addresses are unknown to the network. As a result, broadcast and multihop network topologies may be the best network topology for fully satisfying system requirements.

Instead of using unicast transmission methods, network nodes/vehicles will have to use broadcasting topology. In order to meet the project's safety requirements, the sender of EWM must broadcast the message within the LAN. It is the responsibility of the receiver to determine the response to the individual broadcast message and act accordingly. This can be accomplished through the use of broadcast-oriented and location-based routing techniques.

CPE 1.2 : Correct identification of the EWS message receiver based on proximity data of the sender node

No predefined route is established for the EWM transmission due to broadcasting behavior of the network. Correct identification of the receiving node based on available proximity table data of the sender node has to be evaluated properly.

CPE 1.3 : Creation and maintenance of proximity table from each node

Before the emergency message broadcast can begin, the individual nodes/vehicles must maintain the proximity table information. Because of the LAN network's randomized behavior, this aspect of proximity table creation is critical. The vehicle may have a tendency to change lanes (in case of multi lane scenarios), causing sharp turns, and so on. As a result, the network must update the proximity information of each individual node accordingly after every fixed interval to keep the updated proximity information.

CPE 1.4 : Developing alternate acknowledgement strategy for successful reception of EWM message

The EWM must be transmitted as a broadcast message. Unlike unicast messages, broadcast messages are not acknowledged within the system. As a result, the sender is unaware of the appropriate reception of EWM by the receiver node and may attempt to resend the EWM again, resulting in network redundancy.Thus an alternate acknowledgement strategy has to be implemented to avoid redundancy within network.Upon receiving an alternate ACK the vehicle should stop sending an EWM based on that same even again.

Logistical C.P.E.

CPE 2.1 : Obtaining required physical hardware, software tools and application for system deployment and simulation

Procure the hardware components, hardware resources, technical competencies, and software applications needed for the LAN's deployment, simulation, and performance analysis.

Financial C.P.E.

CPE 3.1: Financial requirements for a scaled-up version of the proposed architecture

To thoroughly test the solution, the project may require financial assistance to acquire licenses/devices that are not available in the telecommunications lab.

Requirements and Constraints

1. The proximity table is location based and maintained for each vehicle which is part of the domain.
2. The proximity table must be kept updated within every 1 second to ensure the integrity of collision detection and effective collision avoidance. In case the proximity table is not updated, the following vehicle might not receive the warning message and there would be no way to keep track of synchronization among the two.
3. The lead vehicle sends redundant periodic broadcast messages to other vehicles in the network until it receives acks from vehicles within its proximity table.
4. The vehicle collision happens within the duration of proximity table updates. If the collision happens in the middle of a proximity table update, there are chances we don't receive acks for the older vehicle(in the proximity table) and the network gets congested due to redundant broadcast messages.
5. The devices used must be able to respond to broadcast messages and decide whether to accept or reject based on the proximity range.
6. Vehicles are not changing lanes during EWM transmission and have limited vision, i.e they are just able to see the vehicle immediately ahead of them.

Design Drivers

Design Driver 1 # : The software Implementation for the proximity data used to update proximity table for each vehicle node should be updated with respect to linear distance between leading and following vehicles. The distance and proximity information should be updated periodically.

Design Driver 2 # : The scaled up version of software should take care of adding a new address (new vehicle) into LAN effectively.

Design Driver 3 # : Software should adhere to traditional wireless network protocols to take care of delay constraints in case of EWM message transmission.

Design Driver 4 # : IImplement a multi hop broadcast environment in which the vehicle broadcasts an EWM and the receiver accepts a message if and only if the EWM is from the vehicle in front of him based on proximity table data.

Design Driver 5 # : Implement a different acknowledgment strategy to avoid network clustering caused by multiple EWM transmissions, i.e. to avoid redundant messages. Unless an acknowledgement is received, the sender must broadcast on a regular basis.

Project Status

Past Planning :

Table below shows the weekly breakdown of what happened in the month of October including analysis and understanding the project definitions.

Week (October)	Status
Week 1 & Week 2	Read papers regarding V2V protocols for vehicle communication
Week 3	Designed the overall flow and how different interfaces communicate with each other
Week 4	Adding more details into the flow diagram(figured out how the overall flow would look like and the important messages that need to be transferred among components) and working on Networking midterm

Future plan:

Table below shows the future plans for building the basic version of software and scaling, testing the same based on data from multiple real life scenarios.

Week (November)	Status
Week 1 & Week 2	Level 1: implementation for creating UDP connections and broadcasting data using these connections Level 2: Implementation of security constraints inside software.
Week 3 & Week 4	Simulating a test environment for executing the test scenarios and validating important test metrics like the performance and scale testing

Software Architecture

The project's software architecture, including functions to be implemented and data to be transmitted between functions, is shown below. As shown in the diagram, the software is divided into two sections, one of which is a while loop and the other is a software section triggered by an event.

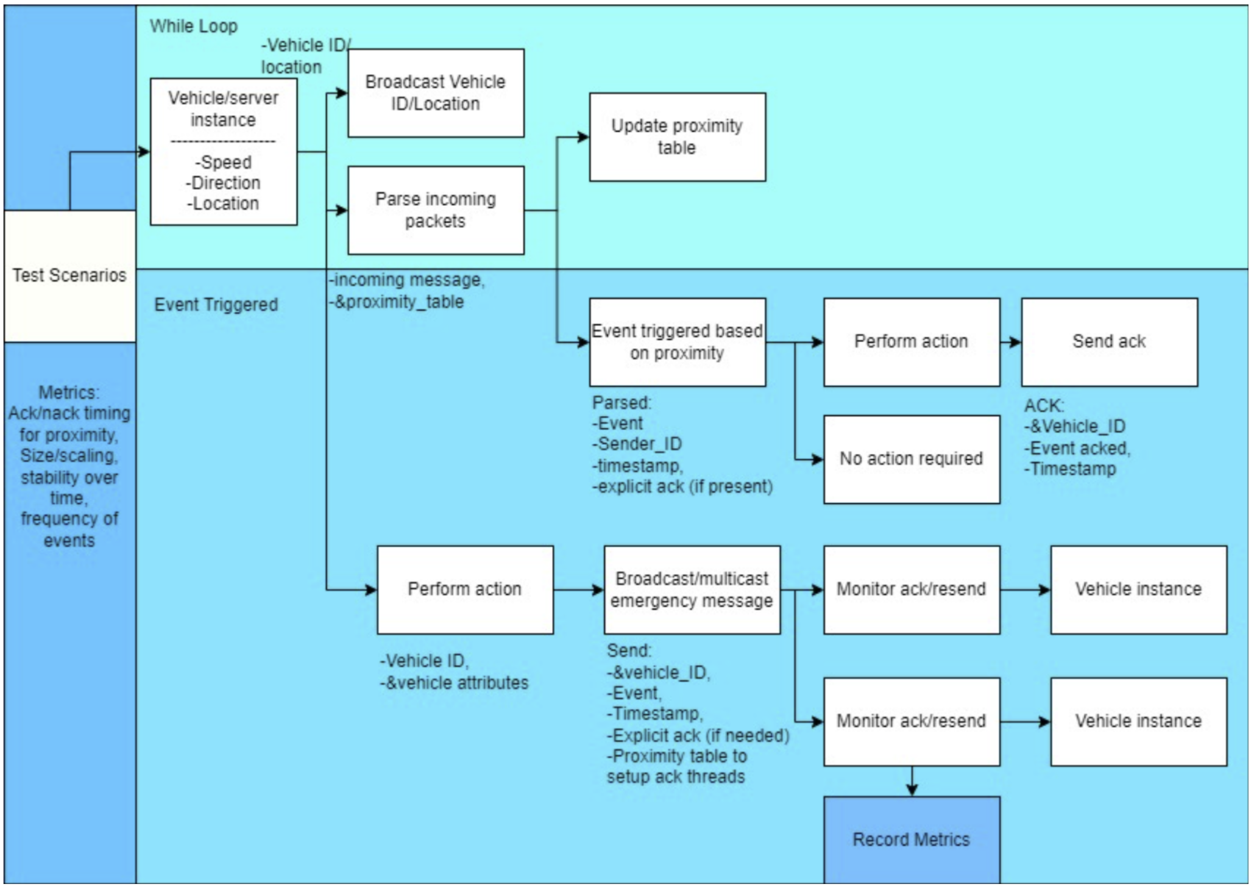


Figure 1 : Software Architecture for vehicle to vehicle broadcast protocol

Function : _Load_Vehicle_Instance_
Function to load vehicle parameters into software. Most of the vehicle parameters like Vehicle Speed, Direction, Location will be loaded from user input. The inputs can be modified to different sets of values to test different scenarios.

Function : _String_Parsing_
After broadcasting the vehicle parameters in a while loop, this function loads the data received in packets from all vehicle instances.Data received from parsed packets will be stored in global data to update the proximity table.

Function : _Broadcast_Vehicle_ID/Location_
This function sends a UDP broadcast message containing vehicle ID, Location, optional: speed, and direction.

Function : _Update_Poximity_Table_
Update proximity table is called on incoming messages being defined as a vehicle broadcast. If the vehicle is within a proximity that requires an ack (say, 100m), the vehicle is added to the proximity table.

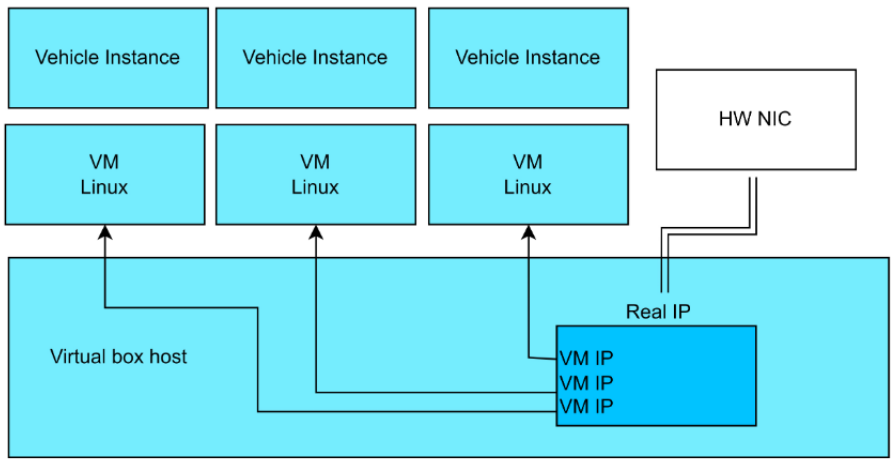
Function : _Event_Trigger_
On receiving a parsed event trigger, the action is performed (decelerate, change direction, etc), and an acknowledgement is sent back to the sending vehicle.
Functionally, this is done through the proximity of the event, but the response is only sent if there is an expectation of response in the message.
Function : _Perform_Action_
Perform Action is initiating an event. The vehicle braking etc triggers a broadcast message saying "this is who I am, and this is what I am doing." There are then separate threads to monitor and target acks from vehicles on the proximity list, or a single thread that copies the proximity list and resends/checks off acks as they come back.

Function: _Update_Location_
For a given time/iteration, the vehicle instance updates it's location based on velocity and direction.

Testing and metrics:
This will be tested over CLI. Ideally, we can automate bringing up vehicle instances and vary parameters/event triggers based on inputs from a test script. The vehicle instances record ack times required, and the test script may iterate time to look into outcomes and system dynamics under varying model parameters. Ideally, the switch could be modified to add in delays etc, but this is beyond the scope of this project.

Results:

Several vehicle instances were set up on a VLAN with a virtual box.



There were 3 instances initiated to show functionality of each part of the code.

```
updating  proximity table
{'e4833bd2-76ab-11ed-810e-b91ea026c07f': 1670471623.6022038}
rand int: 4
emergency event triggered
sending emergency msg, ack table:
{'e4833bd2-76ab-11ed-810e-b91ea026c07f': 1670471623.6022038}
no data recieved: 1670471624.8461943
no data recieved: 1670471625.1778543
no data recieved: 1670471625.7253945
recieved data in event fxn
ack recieved and removed from proximity table, new table:
{}
emergency event ack success
location: 9.909058070182802
updating  proximity table
{'e4833bd2-76ab-11ed-810e-b91ea026c07f': 1670471628.7429996}
rand int: 78
location: 14.24180271625519
```

```
updating  proximity table
{'e2661784-76ab-11ed-ada7-bbfb66963972': 1670471637.460003, 'e4833bd2-76ab-11ed-810e-b91ea026c07f': 1670471637.4602091}
rand int: 4
emergency event triggered
sending emergency msg, ack table:
{'e2661784-76ab-11ed-ada7-bbfb66963972': 1670471637.460003, 'e4833bd2-76ab-11ed-810e-b91ea026c07f': 1670471637.4602091}
recieved data in event fxn
ack recieved and removed from proximity table, new table:
{'e4833bd2-76ab-11ed-810e-b91ea026c07f': 1670471637.4602091}
recieved data in event fxn
ack recieved and removed from proximity table, new table:
{}
emergency event ack success
location: 24.885644149780276
updating  proximity table
```

This code shows 2 moving vehicles pass a parked vehicle and initialize an emergency event each time.

Team Skills and Interests

The table below outlines the skills and interests of the V2V broadcast project team. Also, highlights of the critical project elements (C.P.E.'s) are listed in one of the columns.

Name	Major	Skills / Interests	CPE's
Kiran Jojare	MS(ECE)	Expertise and interest lies in implementation and testing for networked embedded systems. Expertised in vehicle network systems using CAN and CAN-FD. Areas of interest include Wireless Networks, Parallel Computing, Cloud Computing, Embedded System and Autonomous Systems.	1.1 1.3
Bhoomika Singla	MSCS	Worked on software defined networks and utilized L2/L3 concepts in understanding network design and monitoring the traffic flow (number of packets received, transmitted, dropped). Areas of interest include cloud computing, software defined	1.1 1.4

		networks.	
Isha Burange	MS (ECE)	Interests : Network Systems, Computer Architecture and Digital Logic	1.1 1.4
Kevin Jones	MS (ECE)	Work on wireless systems, layer 1 to application layer, and data analysis. I enjoy systems engineering, modeling, and optimization problems.	1.1 1.3

Resources:

Table 1: Required resources to support project requirements

Project Need	Resources	Expert/Technical Consultant
Meeting initial requirements will require a number of vehicle instances to connect over a switch.	https://coding.csel.io/ should be able to provide the compute resources we need	Prof. Jose Santos, Sai Mekala

References

[1] F. Ye, M. Adams and S. Roy, "V2V Wireless Communication Protocol for Rear-End Collision Avoidance on Highways," ICC Workshops - 2008 IEEE International Conference on Communications Workshops, 2008, pp. 375-379, doi: 10.1109/ICCW.2008.77.

[2]Xu, Qing & Mak, Tony & Ko, Jeff & Sengupta, Raja. (2004). Vehicle-to-Vehicle Safety Messaging in DSRC. 19-28. 10.1145/1023875.1023879.

[3]Tan, Chunqing & Qin, & Qin, Gong. (2020). Using Vehicle-to-Vehicle Communication to Improve Traffic Safety in Sand-dust Environment. International Journal of Environmental Research and Public Health. 17. 1165. 10.3390/ijerph17041165.