Smart Vehicular Collison Avoidance/Collision Detection in a VANETs Environement using Wireless Sensor Network

1. Introduction

Over the last decade, the Internet of Things (IoT) has become an increasingly growing subject of conversation worldwide. Many experts believe that by the year 2020 there will be over 25 billion connected devices worldwide. The IoT provides developers with a constantly evolving, smart channel in which to integrate sensor networks with internet technology. The growth is so large that other national standards groups, such as NIST are working non-stop with researchers to develop standards for this new technology. These standards are aimed at promoting the ease in which developers will be able to cross platforms (Velez 2017). For the short time that the IoT has been growing, all of the advances in hardware or software have been proprietary. This means that one company might produce a sensor, and the other a microcontroller, however due to differences in the architectures of either device, they are unable to communicate to one another without advanced modifications.

Smart transducers are low cost, low power, and becoming easier to access and use every day. Utilizing communication between sensor nodes on roadways presents the potential for a number of unique solutions for more effective and efficient management of traffic and traffic information. By replacing the normal sensors with smart sensors, vehicular ad-hoc networks (VANETs) can be generated, and vehicle-to-vehicle communication can be facilitated (Evans 1991).

The paper is organized as follows. A background is presented in section 2 and the experiment setup is discussed in section 3. NS3 simulations are presented in section 4 and the future work and conclusion are given in section 5.

2. Background

A VANET is a wide scoping network which enables vehicle-to-vehicle (V2V) communication directly between vehicles, or between cars and a Road Side Unit (RSU). V2V communication could enable a great number of use cases, mostly in relation to improve driving safety or traffic efficiency but also to provide information or entertainment to the driver. The road side unit might be a stand-alone unit but research is being done on ways to implement the RSU into existing infrastructure. The RSU could potentially facilitate communication between civilians, emergency units, traffic controllers, and even street lights (Drawil 2010 and Kumar 2015).

The VANET classifies collision avoidance solutions based on a number of varying parameters. These parameters may include routing data, collision algorithms, network structures, etc. (Fei 2009). All of this data is combined in some type of complex algorithm. The Kalman filter is one type of these algorithms (Alsbou 2016). The Kalman filter provides a “realistic dynamic model” of a specific event. In case of collision avoidance, the Kalman filter receives various inputs in the form of signals from the vehicle’s sensors and localization information from the VANETs (LeBlanc 2014). The Kalman filter optimizes the problem and provides any driver in range with the best solution or route in order to avoid collision.

A frequency band has been allotted for V2V communications. It is known as the Dedicated Short Range Communication (DSRC) standard. It operates with a frequency of 5.9 GHz and a range of 300 meters. GPS has typically been the standard for vehicle localization; however it still faces accuracy error up to 15 meters (Temkar 2015). These errors are far too large for the safe implementation of GPS into a traffic setting. Using DSRC standards however, sub-meter localization accuracy can be achieved. This is done by using roadside units at known GPS coordinates, combined with inertial measurement and V2V and V2I communication (Zhong 2014).

3. Experiment Setup

Arduino Uno microcontrollers were selected to simulate the receiving and transmitting nodes. The microcontrollers were equipped with various sensors and actuators. The team developed vehicle prototypes using RF controlled cars. LSM303 accelerometers were used to detect the cars motion. HC SR04 ping ultrasonic sensors were used for proximity detection of other vehicles and objects. RF transmitter and receivers were used to transmit data between the nodes. The specific hardware used are listed in Table 1.

Table 1. Hardware used in the experiment

Hardware
Arduino Uno Microcontroller
Adafruit RGB LCD Shield
AdAfruit 10 - DOF Inertial Measurement Unit (LSM303)
Adafruit Ultimate GPS Logger Shield
HC - SR04 Ping Distance Sensor
RF 433 MHz Transmitter Module
RF 433 MHz Receiver Module
DC motor (from car)

The Arduino Uno is a microcontroller board which uses the ATmega328P microcontroller which is a high performance 8-bit microchip. The Adafruit RGB LCD screen was designed to reduce the number of pins occupied by the LCD screen. The shield allows the user to control a 16x2 Character LCD, up to 3 backlight pins and 5 keypad pins using only the two I2C pins on the Arduino Uno board. The Adafruit 10-DOF breakout board allows users to monitor ten different types of motion related data. The IMU combines the following devices: the LSM303DLHC a 3-axis accelerometer and a 3-axis magnetometer, the L3GD20 a 3-axis gyroscope, and the BMP180 a barometric pressure sensor with an on board temperature sensor. The Adafruit Ultimate GPS Logger Shield plugs directly into the UNO. It uses the Digital IO pins 0 and 1. It is designed to log data onto GPS data onto a removable MicroSD card and uses only 20mA, around half the power of other GPS devices. The HC-SR04 ping distance sensor emits an ultrasound wave at 40000 Hz which travels through the air and if there is an object or obstacle in its path it will reflect back to the module. Considering the travel time and the speed of the sound, the module can be programmed to calculate the distance.

The transmitter has a frequency range of 433.9 MHz, and input voltage of 3-12V. The transmitter uses Amplitude-shift keying (ASK) to modulate signals. ASK uses variations in the amplitude of the carrier wave to represent digital data which is considered an inexpensive solution for modulation. The receiver has a frequency range of 433.9 MHz and an input voltage of 5V. DC motors are found in a wide variety of devices. These include high performance electronics, children’s toys, and regular home appliances. The motor in this project is used to drive a remote controlled car. The motor has a supply voltage of 9V. The Arduino script which belongs to this project is designed to control the rotational speed of the motor shaft. The range of speed is from 0 to 210 rpm.

The transmitting node consists of an RC car equipped with an Arduino Uno microcontroller. The Arduino Uno is wired to an LCD screen, an LSM303 Magnetometer/ Accelerometer, HC-SR04 Ping distance sensor, Adafruit Ultimate GPS module, RF 433 MHZ Transmitter Module, and three LEDs. The whole system is connected to the motor of the RC car. The microcontroller is programmed to display compass headings, in degrees, from the LSM303 magnetometer, as well as the distance of any object detected by the HC-SR04 ultrasonic ranging sensors. The microcontroller regulates the car motor's speed based on the impulses from the range sensors. If an impulse is received which is in too close of range, the red led will turn on and the motor shuts off. In addition to stopping, in the event that the car senses something to close, it will also send a message using the RF 433 MHZ Transmitter Module. All of these components in conjunction continually distribute between all of the nodes in the given range. The cars will slow or stop based on proximity and in case of a collision, the exact location of the collision will be transmitted to the surrounding cars with in the transmission range.

The receiving node consists of an Arduino Uno equipped with an LCD screen, an Adafruit Ultimate GPS Module, 3 LEDs, and a RF 433 MHz wireless transmitter. The system turn on a LED if the receiver is not initialized, a yellow LED when the receiver is initialized, and the green LED will flash whenever data is received. The LCD will display the received message on its screen.

4. Collision Avoidance/Detection NS3 Simulation

The Network Simulator NS3 was used to simulate the V2V communications. The first simulation scenario is illustrated in Figure 1 for a two cars model in the presence of a single roadside unit. The figure shows an illustration of the movement of the cars and the messages being sent. For each simulation shown, the red dots represent vehicles and the blue circles represent the transmission ranges or the communication radius which is considered 175 m in our simulation.

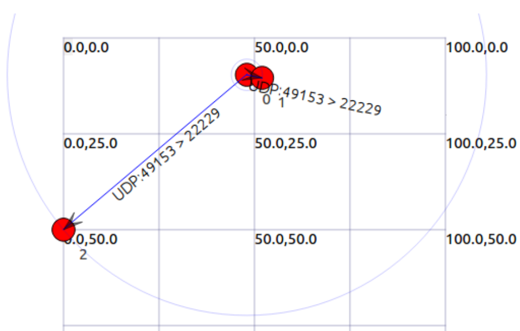

Figure 1. NS3 Simulation of Two Cars Collision in the Presence of a Single Roadside Unit

The simulation was designed such that upon collision, data packets were distributed between all 3 nodes The packets are of size 45 Bytes, transferred at a rate of 3 Mbps and a frequency of 100 HZ. The transmissions were made using IEEE 802.11p protocol with Truncated Exponential Binary Back off.
The second scenario, illustrated in Figure 2, simulated four cars, in the presence of a single RSU, all traveling with constant speeds in different directions. Table 2 shows the transmission of localization data packets for the first 0.0110285 seconds of the simulation.

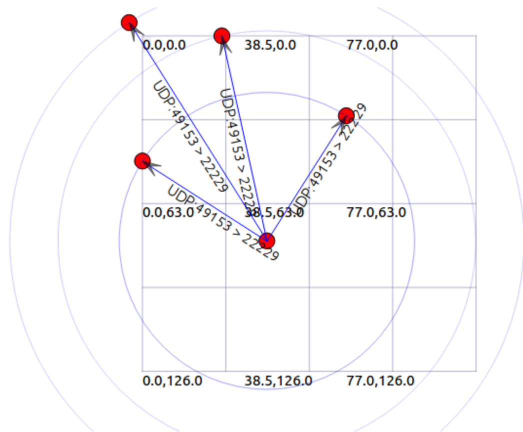

Figure 2. NS3 simulation of a VANET with four cars and one RSU

Table 2. NS3 Simulation output of the time and location of each Vehicle-to-Vehicle communication

From Vehicle	To Vehicle	Time	Location
0	2	0.01012	UDP 49153 > 22229
0	3	0.01012	UDP 49153 > 22229
0	1	0.01012	UDP 49153 > 22229
0	4	0.01012	UDP 49153 > 22229
4	2	0.0105223	UDP 49153 > 22229
4	1	0.0105223	UDP 49153 > 22229
4	0	0.0105223	UDP 49153 > 22229
4	3	0.0105223	UDP 49153 > 22229
1	4	0.0110285	UDP 49153 > 22229
1	3	0.0110285	UDP 49153 > 22229
1	2	0.0110285	UDP 49153 > 22229
1	0	0.0110285	UDP 49153 > 22229

The simulation was designed such that all 4 cars would transmit localization data packets in equal intervals of time, each node will also receive localization data packets. This allows for all nodes to know the location of the other nodes in a given range at the same time. The next step in computer simulation was to simulate actual intersections from which the team could obtain real-world data to compare it to the simulations. The team chose two intersections located on the campus of the University of Central Oklahoma. The first intersection is located at Ayers St. and University Dr., a four way intersection, with stoplights, in Edmond, Oklahoma. The scenario, illustrated in Figure 3 considers four cars in the presence of two RSUs. The scenario simulation is shown in Figure 4.

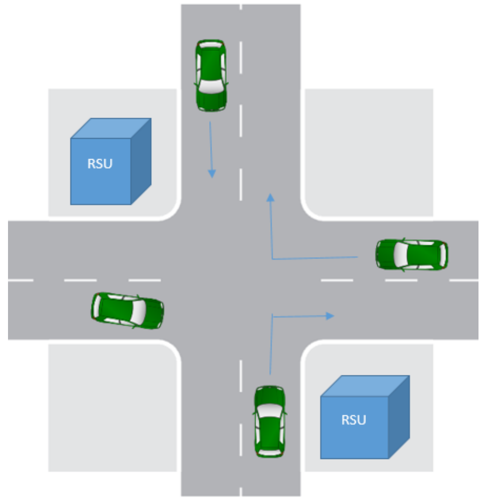

Figure 3. Four way intersection with four cars in the presence of two RSUs

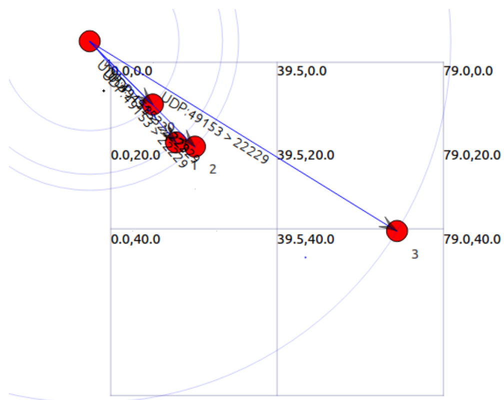

Figure 4. NS3 simulation of four cars in the presence of two RSUs

Table 3 displays the vehicles which sent and received data packets for the first 0.01012 seconds of the simulation.

Table 3. This scenario considers four cars in the presence of two RSUs. The vehicles are at a 4-way intersection on UCO's campus

From Car	To Car	Time (seconds)
0	4	0.01012
0	2	0.01012
0	1	0.01012
0	3	0.01012
0	0	0.01012

The next location to be simulated, illustrated in Figure 5, is a three-way, lighted intersection located at Garland Godfrey Dr. and University Dr. in Edmond. The NS3 scenario simulation is shown in Figure 6. Table 4 displays the vehicles which sent and received data packets for the first 0.01012 seconds of the simulation.

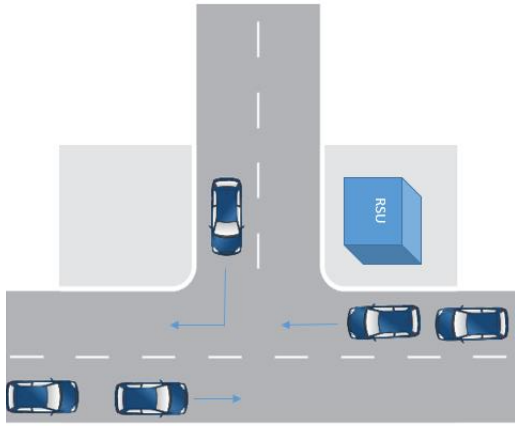

Figure 5.Three way intersection with five cars in the presence of a single RSU

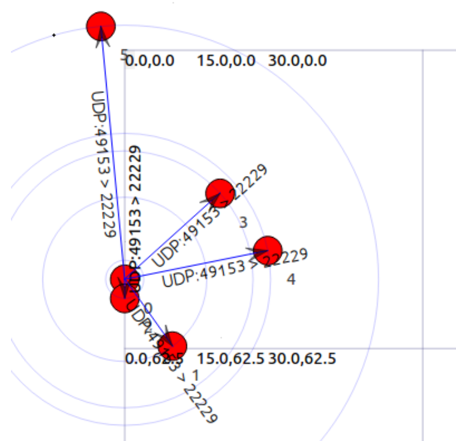

Figure 6. NS3 simulation of five cars in the presence of a single RSU

Table 4 displays the vehicles which sent and received data packets for the first 0.01012 seconds of the simulation.

From Car	To Car	Time (seconds)
0	5	0.01012
0	1	0.01012
0	2	0.01012
0	3	0.01012
0	4	0.01012

The simulation results suggested that localization can be sent quickly and orderly using IEEE 802.11p protocol with Truncated Exponential Binary Back off. In order to maintain stability, the data packets must be small.

5. Future work and Conclusion

Wireless communication technology is rapidly developing. The goal of this research project is to build and simulate a vehicular ad hoc network by sharing messages between sensor nodes to help in collision detection and collision avoidance. This goal was accomplished successfully. Future work will include a possible improvement of the RF controls for the vehicles used and more vehicles need to be introduced into the network. Car-to-Car communication is considered an important technology for the safety of the driver on the road. However, for this communication to be effective, messages transmitted between the cars should be of a small size and should be transmitted quickly and frequently. More simulations will be done considering higher number of cars and different road topologies.

6. References

[1] Alsbou, Nesreen, Heath J. LeBlanc, Edgar Gomez, and Firas Hassan. "Inter-Vehicle Communication Assisted Localization with Resilience to False Data Injection Attacks." CarSys (2016): 15-17.I.

[2] Drawil, Nabil Mohamed, and Otman Basir. "Intervehicle-Communication-Assisted Localization." IEEE Transactions on Intelligent Transportation Systems 11.3 (2010): 678-91. W.

[3] Evans, Gary W.; Carrère, Sybil. Traffic congestion, perceived control, and psychophysiological stress among urban bus drivers. Journal of Applied Psychology, Vol 76(5), Oct 1991, 658-663.

[4] Fei Ye, Matthew Adams, Sumit Roy. V2V Wireless Communication Protocol for Rear-End Collision Avoidance on Highways. Department of Electrical Engineering, University of Washington, Seattle. W.2009

[5] Kumar, K. Prem, S. Jemima Evangelin, V. Amudharani, P. Inbavalli, R. Suganya, and U. Prabu. "Survey on Collision Avoidance in VANET." Proceedings of the 2015 International Conference on Advanced Research in Computer Science Engineering & Technology (ICARCSET 2015) - ICARCSET '15 (2015): n. pag. W.

[6] LeBlanc, Heath J., and Firas Hassan. "Resilient Distributed Parameter Estimation in Heterogeneous Time-Varying Networks." HiCoNS (2014): 15-17.R.

[7] Temkar, Rohini, Vishal Asrani, and Pavitra Kannan. "IoT: Smart Vehicle Management System for Effective Traffic Control and Collision Avoidance." International Journal of Science 4.5 (2015): 3241-246. W.

[8] Velez, J., Trafford, R., DeMartino, A., Tomkins, K., & Topchiy, I. (2017). IEEE Sensors Applications Symposium (SAS) Proceedings, 307-311. Retrieved April 15, 2017.

[9] Zhong, Xunyu, Xungao Zhong, and Xiafu Peng. "Velocity-Change-Space-based dynamic motion planning for mobile robots navigation." Neurocomputing 143 (2014): 153-63. W.